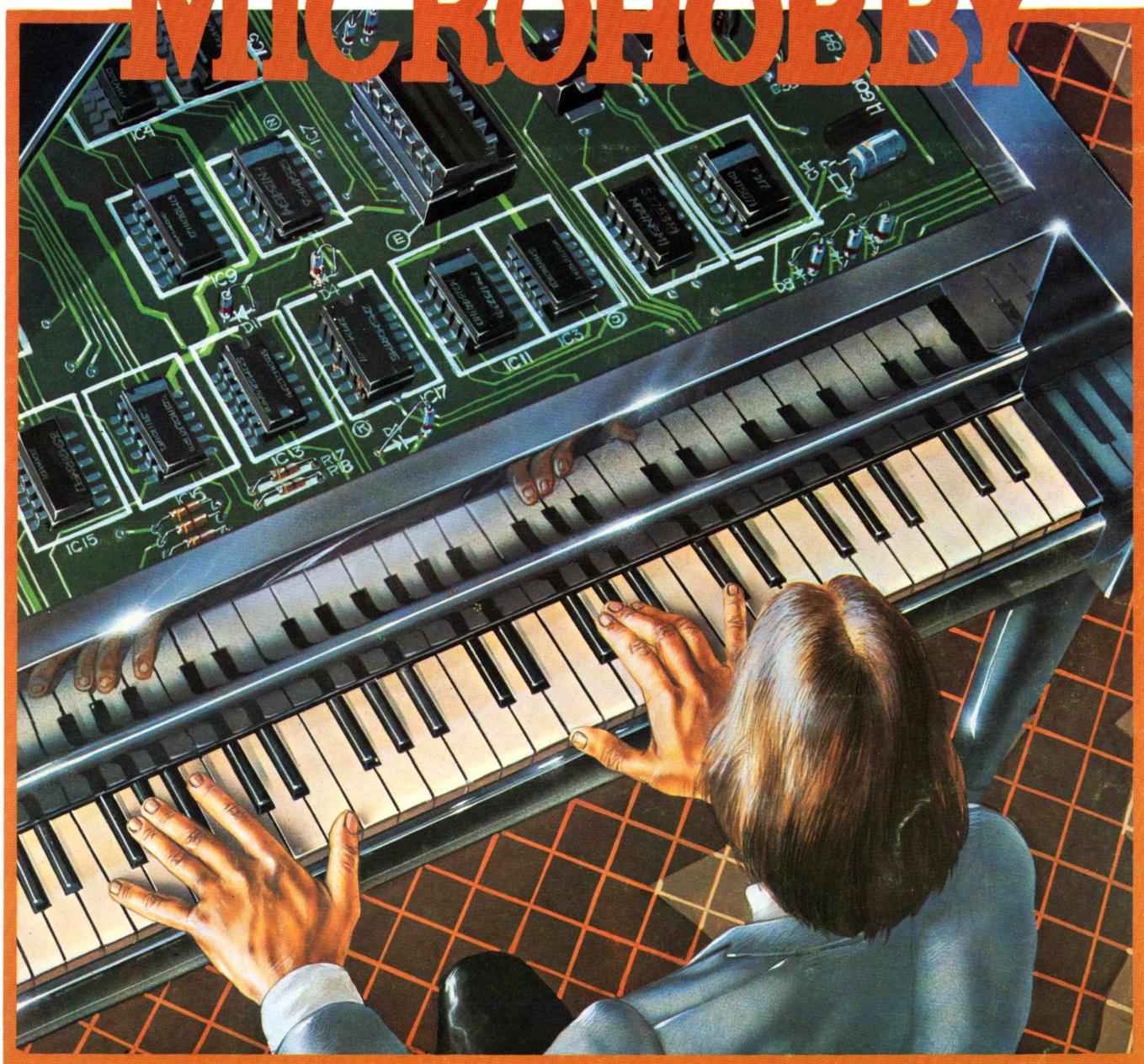
A VOLTA DA SEÇÃO VICE-VERSA

MICROHOBBY

Ano II - Nº 17 - 1985 Cr$ 3.000.

EXEMPLAR DE ASSINANTE — VENDA PROIBIDA

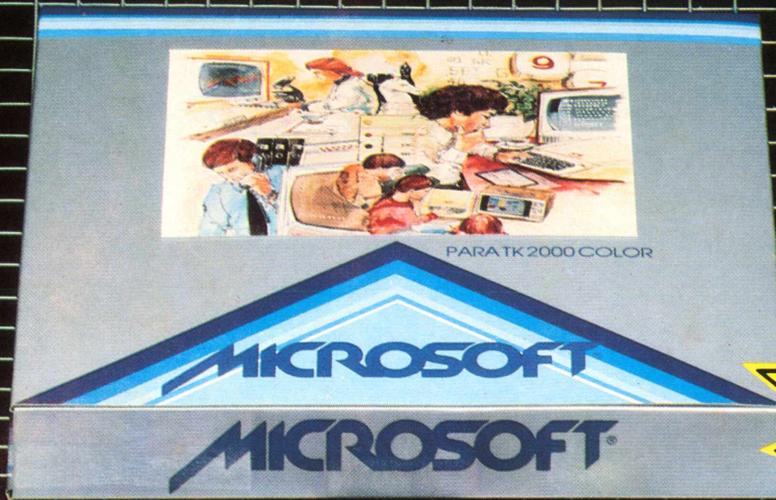
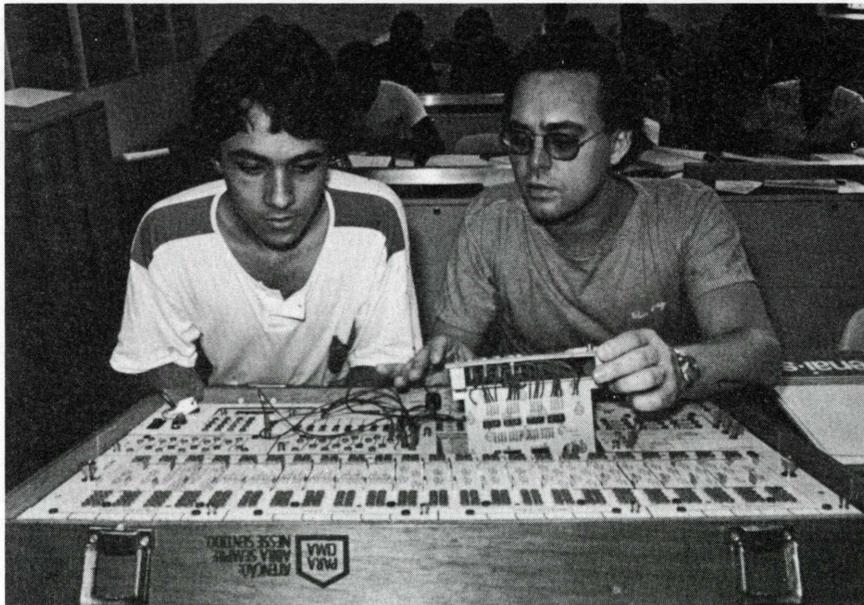
Para Manaus, Boa Vista, Porto Velho, Rio Branco, Santarém, Altamira, Macapá, Rondônia (Via Aérea): Cr$ 3.900.

MÚSICA POR COMPUTADOR

Para o TK 83/85 com 48 K:
MÚSICA POR SOFTWARE
DISCO TK: ARQUIVO PARA SEUS DISCOS

Por dentro do Apple:
NÚMEROS COMPLEXOS

Para o TK 2000:
AS VÁRIAS MANEIRAS DE GERAR SOM

Analisando:
TK WORD PROCESSADOR DE TEXTOS PARA TK 85

Os melhores programas para você.
PARA TK 2000 COLOR
MICROSOFT
MICROSOFT
Garantia integral
MICROSOFT
MICROSOFT
MICROSOFT
MICROSOFT
A Microsoft tem 120 programas em fitas e disquetes à sua disposição. São sistemas aplicativos para acompanhar e agilizar os negócios de sua empresa. E também jogos eletrônicos para você e sua família se divertirem muito. Todos especiais para TK-83, TK-85, TK-2000, Apple II e compatíveis. E todos com a mesma qualidade dos 100.000 programas já vendidos em todo o Brasil.
Procure o revendedor Microsoft mais próximo (se não encontrar os programas Microsoft escreva para a Caixa Postal 54221 - CEP 01000-S. Paulo-SP). Você encontrará os melhores programas da sua vida.
MICROSOFT
Sempre o melhor programa.

Projeto inovador no Senai/SP

Treinador eletro-eletrônico (módulo elétrico) instalado na Escola Senai "Antonio Souza Noschese", Santos.

O Senai vem desenvolvendo, com bons resultados, um projeto de informática que objetiva, entre outros aspectos, atingir a grande demanda da área industrial por técnicos com experiência e conhecimentos da tecnologia digital. Na área de automação industrial a entidade vem se preocupando com dois campos: o desenvolvimento e as aplicações de tecnologias disponíveis.

Diante disto surgiu o projeto de eletrônica visando à especialização da mão-de-obra das indústrias nas áreas de controle de processos contínuos e mecânica de precisão. Para a formação desse pessoal a instituição montou laboratórios, adquiriu microcomputadores, projetou equipamentos e investiu cerca de Cr$ 1 bilhão.

"O projeto é dividido em outros subprojetos envolvendo desde a implantação de cinco cursos de especialização em circuitos digitais e microcomputadores para processos contínuos, à preparação de técnicos em eletrônica para manutenção de periféricos até à construção de receptores de sinais digitais e de um microcomputador didático para aplicações industriais", afirmou John Franklin Arce, assessor técnico da diretoria para assuntos de informática.

A PRIMEIRA EXPERIÊNCIA

A primeira turma-piloto do curso da Lógica Digital formou-se em dezembro do ano passado na Escola da instituição em Santos. Dela participaram cerca de 14 técnicos de diversas empresas que tiveram acesso aos equipamentos desenvolvidos especialmente para o curso — painel demonstrador para lógica digital, microcomputador didático, treinador eletro-eletrônico. Segundo John F. Arce, esta turma-piloto teve um caráter experimental e seus integrantes foram submetidos ao programa do curso, tal como foi idealizado. "Isso permitiu uma avaliação criteriosa do material didático e do próprio ritmo do aprendizado", ressaltou.

A preocupação principal dos organizadores do projeto de eletrônica voltado à formação de técnicos foi com o material didático que, conforme afirmou alguns orientadores da turma, que pretendeu ser o mais pragmático possível. John Arce, destacou a intenção de fornecer ao aluno a situação real que ele vai encontrar na empresa após o curso.

Este ano o Senai já pretende implantar os outros módulos do treinamento (circuitos digitais níveis I e II, comandos lógicos programáveis, etc.). Em março inicia o desenvolvimento definitivo do projeto.

OUTROS PROJETOS

Não só para a automação industrial o Senai está se direcionando. Um dos destaques do projeto é com relação à área didático-pedagógica. Neste setor a instituição pretende desenvolver o ensino de informática mediante a utilização de pacotes aplicativos para controle de produção, além de informações sobre a situação da informática na indústria. A.L.A.

O mercado de informática na opinião dos professores

Embora a oferta de empregos aos profissionais de processamento de dados esteja em expansão, a demanda tende a decrescer sensivelmente, sobretudo no segmento de digitadores e programadores. "O que se vê é o aumento da demanda qualitativa de mão-de-obra na área e não quantitativa", avalia Valdemar Setzer, professor adjunto do Instituto de Matemática e Estatística da USP.

Wilson José Tucci, do Departamento de Computação do Colégio Pueri Domus, destaca que atualmente a maioria dos departamentos financeiros de empresas requisitam profissionais de processamento de dados para comporem seus quadros. O que gera maior procura, segundo Tucci, é que um grande número das pequenas e médias empresas sentem a importância de entrarem na era da informática para se tornarem mais competitivas.

Apesar da difusão de programas aplicativos, "pacotes" prontos disponíveis, na opinião de Tucci, a maioria das empresas estão optando, geralmente, por programas personalizados, desenvolvidos internamente com pessoal próprio.

Outro trabalho importante realizado pelos profissionais de processamento de dados, segundo Tucci, é a manutenção do software. "Num país inflacionário como o nosso, um controle de estoques, por exemplo, deve ser corrigido quase diariamente", explica.

Ele falou também sobre o aparecimento das linguagens de Quarta Geração e na facilidade que traz aos usuários finais na construção de seus próprios programas, definindo-as como um instrumento que não constitui ameaça aos analistas. Tucci esclarece que, a partir dessas linguagens, os analistas vão poder fazer realmente análise e deixarão as atividades paralelas de programação, alheias à sua área. Para ele, com a ajuda dessa ferramenta, o usuário saberá melhor o que quer, possibilitando ao analista adequar essas linguagens à suas necessidades reais, além de continuar a dar, também, a manutenção ao software.

Aviso aos Navegantes

Olhando sob um outro ponto de vista, Valdemar Setzer vê com pessimismo a oferta de empregos na área. Para ele, o que já está acontecendo hoje é uma sensível diminuição da demanda de mão-de-obra, principalmente dos digitadores e programadores. "Essas profissões foram criadas há pouco tempo e também vão desaparecer rapidamente", afirma.

Ele destacou como um "aviso aos navegantes", em processamento de dados só vão sobrar os analistas versáteis e muitos especializados em várias áreas, para terem a possibilidade de mudar quando forem substituídos por softwares específicos.

O mercado de trabalho, para Setzer, ainda é fraco em nível, isto é, requisita profissionais, mas ainda não há uma exigência de maior qualidade. A curto prazo, ele prevê que a oferta de empregos diminuirá, mas subirá de nível exigindo, proporcionalmente, profissionais mais gabaritados em suas áreas.

Ele prevê também, nesse processo, o desaparecimento do CPD das empresas. Para Setzer, a máquina vai ter uma necessidade mínima de operadores e digitadores, já que todo o processamento será feito on-line. Ele explica que antes, o CPD recebia tarefas e produzia relatórios para a empresa. Agora, com a entrada da operação dos usuários finais e com a difusão dos micros, esse sistema está ficando cada vez mais obsoleto, pois o próprio usuário, com as ferramentas que lhes são dadas hoje em dia, pode facilmente manipular a máquina e programá-la para seu próprio uso, sem a necessidade de requisitar relatórios ao CPD.

Formação Profissional

O baixo nível da formação de tecnólogos de computação foi outro assunto abordado por Valdemar Setzer. Para ele, os profissionais devem ter, em primeiro lugar, uma boa base matemática, o que geralmente não ocorre nesses cursos. O que ele sugere é o Bacharelado em Ciências de Computação, como a melhor formação da área atualmente. A procura desses cursos, hoje, sobe ano a ano. Setzer informou que em 83 haviam 4 mil candidatos, em primeira opção, para apenas 36 vagas do Instituto de Matemática da USP.

Ele criticou também a maneira como o jovem escolhe a informática como profissão, salientando que "não basta gostar de fazer joguinhos eletrônicos". Setzer aponta, principalmente, a necessidade de se gostar de matemática e lógica. Outro erro freqüente entre os profissionais é entrar nessa profissão, só porque acha a máquina interessante.

Segundo Setzer, não se aprende computação via cursos de linguagem, mas sim, através da matemática e do conhecimento de algorítimos; "a lógica que está atrás de todas as linguagens", explica. **A.L.M.**

Videogames, a grande presença na Hobby & Lazer

Israel Teixeira

Em meio a um ambiente típico de "crie e faça", a informática esteve presente na feira de Hobby & Lazer, realizada em São Paulo, através de três empresas: Midrodigital, CCE e Dynacom. Estas levaram ao Anhembi, durante aproximadamente 15 dias, seus micros, videogames e, principalmente, os jogos animados, que cativaram a atenção de crianças, adolescentes e seus pais.

Conforme salientou Carlos Alberto, assessor de vendas da CCE, este tipo de feira serve para que as empresas fabricantes apresentem aos revendedores de seus produtos os últimos lançamentos, assim como possibilita ao consumidor final o acesso ao equipamento do qual ele poderá vir a ser um usuário.

As três empresas tiveram seus "stands" bastante freqüentados durante a realização da Hobby & Lazer. E, entre os produtos que mais chamaram a atenção do público estavam realmente os videogames. Entre as marcas apresentadas estava o Onyx, um dos últimos lançamentos de final de ano da Microdigital. Segundo os diretores da empresa, o videogame veio atender à saga dos usuários, já que a expectativa pelo lançamento do Onyx no mercado era muito grande.

O Onyx JR é compatível com o sistema Atari e apresenta uma inovação ressaltada pelos representantes da empresa: a tecla "pause", que permite o congelamento automático de qualquer cena desejada do jogo.

Já a CCE apresentou seu último lançamento — o micro MC-1000 — e o disco laser mostrado pela primeira vez na feira. **A.L.A.**

Servimec expande seu campo de atuação

Programa institucional de informática (Micro-mirim).

Dando prosseguimento à expansão iniciada no ano passado, quando inaugurou unidades educacionais em Santo Amaro e no ABC paulista, a Servimec fechou 84 com um crescimento real de 30% (faturando aproximadamente Cr$ 9.600 bilhões), e lançou em janeiro, as sementes para a implantação da Rede Nacional de Ensino e Serviços de Informática, Renesi.

A primeira unidade da rede/Servimec foi inaugurada no Rio de Janeiro associada a duas outras empresas: a Negócios Brasileiros de Informática — NBI e a Planejamento, Projetos e Sistemas, PPS. O objetivo dos diretores da empresa é ligar as principais capitais do Brasil On-Line com a sede em São Paulo, para fornecer à comunidade local, serviços (bureau, comercialização de equipamentos, consultoria, planejamento e assessoria), educação e ensino.

Na primeira unidade da Renesi funcionarão 10 microcomputadores I-7000 12 TK-85, seis terminais de vídeo Burroughs ligados à central, via linha telefônica — privada — e um terminal impressor; além de 20 professores e uma equipe de psicólogos. A empresa instalou quatro salas de aulas, um show-room que terá várias marcas de computadores em demonstração. O setor de prestação de serviços em consultoria e assessoria (o bureau de serviços) só estará em funcionamento a partir do segundo semestre, pois até lá estará em fase de implantação, com o intuito de expandir a base instalada de equipamentos, contratando no momento, computadores IBM.

O objetivo da rede/Servimec, segundo Antonio Barrio Junior — diretor superintendente, é levar à comunidade local os recursos tecnológicos e estruturais dos grandes centros (por exemplo, São Paulo), através de microcomputadores e terminais interligados com as principais cidades do país. Para isto, a empresa oferecerá na área educacional, cursos como COBOL, BASIC e BASIC Avançado, programas institucionais (como BASIC-Mirim, BASIC-Mulher, entre outros) e seminários técnicos para profissionais.

A SEDE DA PRIMEIRA UNIDADE DO RENESI

A instalação da primeira unidade no Rio de Janeiro gerou, conforme afirmou Barrio Junior, um investimento de aproximadamente Cr$ 250 milhões e a previsão de faturamento está estimada em torno de 25 a 35 mil ORTNs.

Para conseguir esta meta a empresa instalou a sede no centro do Rio e preocupou-se, segundo Antonio B. Junior, em fornecer aos alunos o mesmo apoio e recursos didáticos, inclusive o estágio com registro em carteira para os melhores alunos (oferecidos pela empresa nas escolas de São Paulo).

Na área de serviços, a Servimec pretende utilizar a rede com o sistema "timesharing", valorizando o uso do micro a nível local para depurar informações e a utilização dos terminais Burroughs para a transmissão de informações com grande base de dados.

Os cursos no Rio de Janeiro já estão recebendo inscrições e as aulas serão ministradas em três períodos: manhã, tarde e noite. O curso de COBOL tem duração de 6 meses, o de BASIC 30 horas além das aulas práticas nos equipamentos disponíveis. **A.L.A.**

ERRATAS

Reportagem Especial (edição nº16)

— O nome correto do diretor de divulgação da SUCESU/SP e da ADP Systems é *Edes Landim* e não *Edes Laudim.*

— No sétimo parágrafo da matéria "Graduação e pós-graduação em Informática" leia-se: *Na USP (curso de bacharelado em informática), era, igualmente, ao de UFRJ, até o ano passado, subordinado ao de Matemática.*

— No Box — Uma proposta inovadora: a universidade invisível — o nome correto do diretor da Apple Cursos é *Eduardo Previdelli* e não *Eduardo Prividelli.*

CALENDÁRIO

11 a 13/2 — **Como projetar, desenvolver e operar sistemas de dados** — Seminário da Compucenter. São Paulo, SP. Telefone: (011) 255.5988.

24/02 — **Digitação** (20 horas) — ADP Systems. São Paulo. Telefone: (011) 227.4433.

25/02 — **DOS/VSE** — Idem Acima.

25/02 — **BASIC em microcomputadores** — ADP Systems — Idem acima.

11 a 15/3 — **Engenharia da Informação** — Seminário Compucenter.

20 a 23/3 — **Micro-Festival/SP** — Guazelli Associados. Informações: (011) 285.0711.

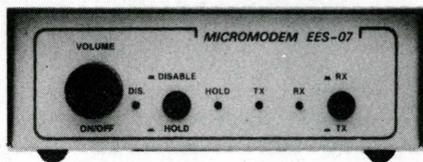
28 e 29/3 — **Tecnologia de Combustão** — Seminário da ABACE — Associação Brasileira de Administração e Conservação de Energia. Telefone: (011) 285.2490.

25/03 — **Básico de Programação IBM** — ADP Systems.

25/03 — **COBOL IBM** — ADP Systems.

11 a 14 de março — **Wordstar** — carga horária: 12 horas. Compushop — São Paulo. Informações: (011) 815-0099/852-3366.

26 a 27/3 — **LOTUS 1, 2, 3 e Introdução ao PC** — carga horária: 16 horas. São Paulo. Informações: Idem acima.

TKs-85 já podem conversar

Um modem desenvolvido por um jovem engenheiro eletrônico, que permite a conversação entre os micros TK-85, via linha telefônica, está fazendo a cabeça de seus usuários. O EES-07 foi desenvolvido por Antonio Paulo Hawk nas suas horas de folga devido a sua frustração em não poder ter acesso a interfaces que lhe permitisse tal feito.

Antonio Paulo faz questão de dizer que o micromodem não foi projetado, pois ele aproveitou o que já estava pronto. No início, tentou passar os sinais diretamente sem mexer no sinal digital, mas depois percebeu que seria melhor utilizar os recursos internos do micro. "Com a saída para fita cassete surgiu a possibilidade de aproveitamento do sinal modulado existente, bastando apenas balanceá-lo, filtrá-lo e jogá-lo na linha telefônica", ressaltou. Como a linha telefônica já é balanceada, o invento de Hawk serve como um instrumento de comunicação entre os TKs distantes, eliminando a interferência de ruídos e tornando viável a transmissão dos sinais digitais junto com os elétricos.

A preocupação central de Antonio Paulo no desenvolvimento do modem foi com o custo final do periférico, para que este não fosse repassado ao usuário final de um micro como o TK-85. Assim sendo, o engenheiro investiu na produção de um lote de 300 peças, sendo que a partir da publicação de uma matéria em um grande jornal de São Paulo, já foram vendidas, por telefone, 60 peças. Atualmente o micromodem está sendo comercializado nas lojas da Áudio a um preço de 10 ORTNs.

Trocando programas via telefone

O funcionamento do EES-07 é bastante simples. Conectado à saída do gravador e ligado na linha telefônica, o modem permite ao usuário a troca de programas utilizando apenas, além dele, um gravador para a recepção (através da instrução "hold" do aparelho e LOAD do micro) e entrada de dados (RX do modem e SAVE do TK), além da utilização de funções e programas especiais.

Mas, segundo o inventor, o micromodem pode ser utilizado também para conversação entre micros de outras linhas, que possuem saída para gravador, como o TRS-80 e Apple, dependendo porém da compatibilização do formato dos sinais emitidos, efetuada por Hawk em sua empresa de serviços de nacionalização e transformações de equipamentos na área de eletrônica.

Hawk acredita que seu invento será bastante difundido a partir do momento em que os grupos de usuários — como os clubes de micros — descobrirem sua grande utilidade. Para ele, o EES-07 abre perspectivas para um maior intercâmbio entre os usuários deste tipo de micro. "Um clube de micros com mil programas tem grande campo de atuação com o modem, devido ao seu baixo custo, facilidade de manuseio e aplicações para a troca de programas entre os usuários, por exemplo", complementou.

Outros projetos

Sempre pensando em um entrosamento entre os usuários, Hawk chegou à idéia de desenvolver outros periféricos que pudessem permitir a estes, acessar serviços como o Cirandão e o Videotexto.

Para isto, ele desenvolveu uma interface baseada em um modelo inglês que permite a usuários do TK e outros micros, acessar o serviço Cirandão da Embratel — Empresa Brasileira de Telecomunicações.

Antonio Paulo Hawk já montou a interface, porém a maior dificuldade que tem encontrato é com relação à importação do circuito integrado. Conforme afirmou, os únicos caminhos encontrados por ele, para conseguir o CI, foi através de contrabando.

"A interface obedece o mesmo plano de desenvolvimento do micromodem e seguirá a mesma preocupação: ser acessível ao pequeno usuário, principalmente em termos de custo". **A.L.A.**

Sistemas de Vendas da Xerox

PSS II — **Professional Selling Skills** é o primeiro de uma linha de 19 produtos a serem comercializados pela Xerox e que apresentam uma imagem diferente da empresa no mercado brasileiro. O PSS II é um sistema de treinamento em vendas, envolvendo administração e gerenciamento em vendas e será representado no Brasil através de uma empresa subsidiária, a TDX/Xerox.

A TDX vem atuando no mercado de treinamento de pessoal nos Estados Unidos há aproximadamente 20 anos e, conforme disse Frederico Luiz Oliveira Lima, gerente de operações comerciais, das 500 maiores empresas citadas pela revista Fortune americana, pelo menos 422 utilizam o sistema da Xerox.

"O PSS II é formado por uma série de técnicas de vendas do mais alto nível e a maior prova é o know-how do nome Xerox", afirmou Frederico.

O sistema estará à disposição das empresas nacionais, segundo Frederico Lima, de três formas distintas: seminários abertos e fechados (as empresas contratam os serviços da TDX/Xerox e esta envia seus instrutores para o treinamento dos respectivos profissionais) e através da venda de "kits" de treinamento, fornecendo assim, a tecnologia de vendas da empresa. Desta forma, a Xerox treina os futuros instrutores dos profissionais das empresas. **A.L.A.**

EDITORIAL

O uso do computador como meio de expressão artística é um assunto que pode gerar muita polêmica.

Será que o que o computador produz é arte? Recentemente ouvi um disco de música composta por computador, mais precisamente por um IBM 370. Isso me pareceu uma massa sonora formada por sons distribuídos de maneira muito artificial. Entretanto, estas músicas tinham um mérito: de quando em quando, uma ou outra frase me parecia um achado musical que, infelizmente, logo sumia no meio de um mar de sons.

Recentemente discutimos a possibilidade do computador pensar. Agora o problema reaparece de forma diferente. O computador é capaz de produzir uma obra de arte?

Essa pergunta precisa ser bem examinada. O computador pode ser usado como um meio ou um instrumento de produção artística, tal qual uma câmara fotográfica. Neste caso, o computador substitui os meios tradicionais de expressão: o pincel do pintor, o teclado do pianista ou a máquina de escrever do escritor. É neste sentido que trabalhamos para produzir este número, elaborando alguns programas que transformam o computador num instrumento musical, como um piano eletrônico ou uma caixa de música.

Uma outra forma de usá-lo é como uma "fonte de inspiração".

Tomemos o exemplo das músicas compostas pelo IBM 370. Se o seu conteúdo genérico é duvidoso como obra de arte, certos trechos, quando localizados em outra música, podem ser expressivos. Ou seja, um compositor pode usar o computador para simplificar seu trabalho intelectual de procura da frase musical adequada. Pode, num momento em que um trecho musical de sua autoria "pede" uma frase complementar, ouvir um número de frases compostas ao acaso pelo computador e escolher uma delas que mais se adapte ao seu trabalho.

O terceiro modo é transformar o computador num artista. Aqui é que entramos num terreno perigoso. Entretanto, de uma forma geral, é o mesmo problema que enfrentamos ao decidir se uma obra vanguardista é ou não arte. De qualquer maneira, acreditamos que a arte ainda é um reduto humano e que um computador, por não ser humano, não pode produzí-la. E se um determinado número de pessoas julgar que uma obra produzida por um computador, sem conhecer este fato, é arte?

O que eu pretendo com este Editorial é lançar uma dúvida na cabeça do leitor. Isso gerará uma atitude crítica e quem se detiver em fazer conjeturas talvez não se surpreenda em ver um concerto para computador e orquestra, escrito pelo próprio computador ou, pelo contrário, se isso nunca ocorrer.

Alvaro

Índice

Expediente

DIRETOR RESPONSÁVEL
Szaya L. E. Seifert
GERENTE GERAL
Dijalma Peinado
Marcia Regina Dominiquini (assistente)
EDITOR
Álvaro A. L. Domingues
JORNALISTA RESPONSÁVEL
Ana Lúcia de Alcântara — Mt. 14.495
EDIÇÃO DE NOTÍCIAS
Ana Lúcia de Alcântara
REDAÇÃO E ANÁLISE
Rogéria Gomes da Silva (secretária)
Ana Luisa Mahlmeister (colaboradora)
Solange Aparecida Menezes (revisão)
ASSESSORIA TÉCNICA
Paulo Lauand
Wilson José Tucci
Aroldo Possuelo Carvalho
Angel D. Zaccaro Conesa
DIAGRAMAÇÃO
Esteves Propaganda
CORRESPONDENTES
Rio de Janeiro — Fátima França
PUBLICIDADE
Aurio José Mosolino (supervisor)
Eduardo Garcia de Souza
ASSINATURAS
Siumara Farisco
Marcos Lorenzi
CIRCULAÇÃO
José Aparecido Bueno
DISTRIBUIÇÃO
Fernando Chinaglia Distribuidora S/A
COMPOSIÇÃO E FOTOLITOS
Ponto Reproduções Gráficas Ltda.
IMPRESSÃO E ACABAMENTO
Editora Parma Ltda.

MICROHOBBY é editada mensalmente por MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO LTDA., INPI 2992 Livro A
Endereço para correspondência:
Caixa Postal 54096 — Fone: 255-0366
CEP 01296 — São Paulo, SP
Para solicitar assinaturas (12 números) envie cheque nominal à MICROMEGA P.M.D. Ltda., no valor de Cr$ 30.000

MICROHOBBY 17

Réu confesso na Investigação Policial

Caro Editor,

Aceitando sua sugestão, dirijo-me à seção de Cartas com algumas correções ao programa "Investigação Policial". O objetivo é, de um lado, colaborar com a impressão que saiu defeituosa, dificultando ao leitor — principalmente ao que se inicia ou tem preguiça de deduzir — e quanto às modificações introduzidas à revelia de seu autor.

Assim sendo, no programa II cabem as ressalvas:

Figura 1

(cancelar as linhas 530 e 550 do programa impresso)

Figura 2

O leitor atento notará que, além de detalhes de mensagens que este autor faz questão que sejam mantidos, os laços de tempo (FOR-NEXT) foram reduzidos na revisão da Redação. Mantenho que — embora eu mesmo seja treinado em Leitura Dinâmica e espante meus colegas pela velocidade com que leio e apreendo um texto — devemos colocar tempos mais longos no vídeo (um meio diferente do papel impresso), ao qual a maioria dos nossos "consumidores" de programas — os amigos, parentes, colegas — não está acostumada a ler rapidamente. Note ainda o leitor a linha 360, corretamente substituída pela Redação, porém fugindo ao meu objetivo: introduzir o leitor a este detalhe de lógica Booleana (IF NOT), tão pouco empregado, e a nosso dispor no teclado dos micros compatíveis com o TK 83/85.

Aproveitando este contato com os leitores — além daquele através de meus programas publicados neste veículo — para externar de público meu agradecimento à equipe produtora de Microhobby, ao apoio e simpatia de seus integrantes. O leitor não desconfia — nem de longe! — a grande sensibilidade que se esconde detrás dos bits, bytes, FOR, INKEY$ e o mais que segue dessa simpático grupo.

Deixem-me aproveitar para desejar a todos um fantástico ano de 1985, pleno de SAVE, PLOT, RUN e LOAD, com um mínimo de GOSUB, RETURN e PAUSE!

Nilson Martello — São Paulo SP

```
  60 LET T=INT (RND*15)+6
 320 IF B=1 AND S=A OR B=3 AND S
=A THEN PRINT "ENCONTROU A ARMA.
"
 330 IF S=A THEN LET AA=1
 340 IF B=2 AND S=D OR B=3 AND S
=D THEN PRINT "ENCONTROU AS DIGI
TAIS."
 350 IF S=D THEN LET AD=1
 360 IF NOT AA AND NOT AD THEN P
RINT "NAO HA NADA AQUI..."
 370 FOR N=1 TO 150
 520 GOTO 700
```

Figura 1

```
 640 PRINT AT 10,14;" RAT ";AT 8
,9;" TA "
 730 PRINT AT 10,5;"DESEJA OUTRA
 AVENTURA ?","(S/N)"
 760 PRINT AT 15,0;"ENTAO MEUS I
NTEGRADOS","AGRADECEM..."
 770 FOR N=1 TO 150
1010 FOR N=1 TO 120
1040 PRINT AT 8,0;"SE V. ENCONTR
AR A TEMPO A ARMA DO CRIME E A
S IMPRESSOES DIGITAIS"
1050 FOR N=1 TO 150
1060 NEXT N
1070 PRINT AT 11,0;"O CRIMINOSO
SERA' APANHADO.POREM CAUTELA:ELE
 ESTARA' DE OLHO EM VOCE..."
1080 FOR N=1 TO 180
```

Figura 2

Caro Martello,

Sua dedução estava correta e o culpado pelas modificações aqui se apresenta. Embora as modificações que fizemos o tenha desagradado (e com razão), nosso intuito foi apenas melhorar (boas intensões são atenuantes?). Publicamos sua carta aqui para que nossos leitores tenham oportunidade de verificar como você concebeu originalmente o jogo antes de nossa intervenção.

Por outro lado, em uma próxima oportunidade, quando julgarmos necessário alguma alteração, entraremos em contato primeiramente com você. Desta forma, evitaremos que alguns de seus cabelos sejam arrancados.

Quanto a humanidade por trás da Microhobby, julgamos que você, com seus artigos e programas, também contribui para a humanização dos bits e bytes que compõe as páginas desta revista.

O Editor

TK 2000, TKS 800 e Sugestões

Sou assinante da Microhobby desde o número 2 e tenho observado a constante melhora da revista a cada mês. Parabéns.

Tenho algumas dúvidas e sugestões que gostaria de expor a vocês e, eventualmente, a outros leitores da Microhobby.

Dúvidas:

1) Dentre o TK 2000, o TK 2000 II e o TKS 800, qual o melhor? Obviamente para esta comparação, contariam os preços, a memória, resolução, etc.

2) Programas feitos para o TK 83 rodam no TK 2000 e/ou no TKS 800?

Sugestões:

1) Poderiam fazer duas seções mensais com o seguinte conteúdo:

— análise de novos computadores (Comodore, Macintosh)

— a história da Computação.

2) Acho também que seria melhor publicar mais programas aplicativos domésticos.

3) Também seria importante, na minha opinião, que fizessem uma tabela de conversão para as funções dos seguintes computadores: TK/APPLE/TRS 80/Sistema 700.

Paulos Cezar P. R. Camarion
Rio de Janeiro RJ

Caro Paulo Cezar,

Agradecemos primeiramente os elogios que nos faz. Quanto aos computadores TK 2000 II e TKS 800, temos a informar que:

a) O TK 2000 II é inteiramente compatível com o TK 2000, com a diferença básica de que o TK 2000 é apresentado em três versões: 48, 64 e 128 k e os modelos com menos memória podem ser expandidos até 128 k.

b) O TKS 800 é compatível com o Color Computer da Radio Shack.

c) Ambos os computadores têm capacidade para trabalhar com gráficos coloridos, som e alta resolução.

d) Não há possibilidades de se indicar o melhor, pois ambos são destinados a públicos diferentes. O TK 2000 II tem a possibilidade de admitir um disquete, expansível até 128 k e admite apenas o sistema operacional TKDOS (bastante semelhante ao DOS 3.3 do Apple). O TKS 800 admite até quatro disquetes e roda, além de seu sistema operacional, o CP/M.

Diante disto, o melhor que você tem a fazer é examinar ambos os equipamentos "ao vivo" ou aguardar a análise que faremos de ambos os computadores em breve e, então, escolher qual o melhor que lhe convém.

Uma de suas sugestões já foi realizada: nas edições 15 e 16 falamos sobre o Macintosh, com detalhes, na seção "Por Dentro do Apple".

Programas aplicativos surgirão sempre que possível. Aguarde!

A seção Vice-Versa tratará das traduções de comandos e instruções de um computador para outro. Uma simples tabela de conversão não basta, em virtude da diferente organização de tela, memória, uso de instruções, etc.

TKS 800 e TK 2000 II e A Volta do Barão Vermelho

Gostaria de parabenizá-los pelo ótimo trabalho desenvolvido até hoje por vocês da Micromega. Sou leitor assíduo desta revista e espero que continuem assim, ensinando e auxiliando todos os TKmaníacos, como eu.

Gostaria de saber o seguinte:

1) Soube do lançamento do TKS 800 e que um novo TK 2000 irá entrar em ação. Como será este computador?

2) De onde e como surgiu a idéia da Microdigital colocar as iniciais TK em seus computadores?

3) Rodei o programa "A Volta do Barão Vermelho" e pouco depois de atingir o alvo, meu TK "saiu do ar". O que houve?

Wladimir Pereira Batiston
Campinas SP

Caro Wladimir,

Como já dissemos, o TK 2000 II é fundamentalmente o mesmo TK 2000, só que apresentado em três versões: com 48, 64 e 128 k, com possibilidade de expansão de memória (até 128 k). O aspecto geral da máquina será o mesmo, mudando apenas a cor.

O nome TK vem de um de seus criadores, o sr. Thomas Kovari, atualmente um dos diretores da Microdigital.

Quanto ao Barão Vermelho, nós o testamos e não tivemos problemas. Verifique se você não cometeu nenhum erro de digitação e confira os códigos em linguagem de máquina. Deve ter havido algum engano de sua parte. Por outro lado, conferiremos novamente este programa. Caso haja qualquer coisa errada, nós voltaremos a publicá-lo.

Capitão Gancho, Mazogs e Simulado de Vôo

(. . .) Bom, escrevi principalmente por um motivo: li nos números 5 e 6 da revista Microhobby o artigo: "Proteção Jurídica de um Programa de Computador" e notei que três empresas que trabalham com software para computadores compatíveis com o TK 83/85, estão comercializando os mesmos programas, embora com nomes diferentes.

Eu queria saber se isto é proibido ou não. Quem foi o primeiro fabricante a comercializar o Mazogs? Este software está sendo comercializado por três empresas, cujo nome omitirei.

Aproveitando o momento, gostaria de perguntar duas coisas:

1) No programa Mazogs, quando já tenho o tesouro e estou em busca da saída, eu posso comprar uma espada?

2) No Simulador de Vôo, às vezes, quando o avião cai, surge a mensagem: "Pôs o avião em *stall*"

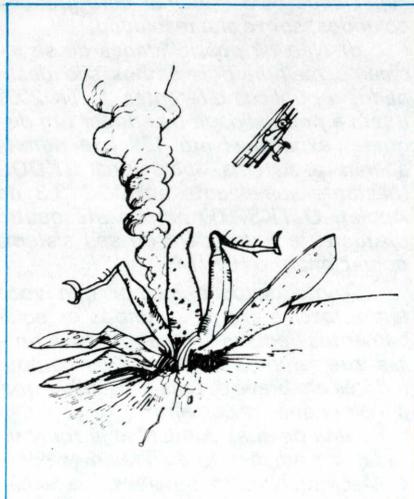
O que isto significa?

Fábio Ferreira de Paulo
Osasco SP

Caro Fábio,

A origem do programa Mazogs é inglesa e as firmas que o comercializam fizeram adaptações e modificações significativas no software. No momento que isto ocorreu (tradução e adaptação do software), não havia uma legislação específica sobre software no Brasil, de forma que nenhuma das empresas é detentora dos direitos de exclusividade de fabricação deste programa, de forma que não se configura a pirataria entre elas, já que a fonte é a mesma. Além disso, pelo menos a versão que conhecemos e comentamos na revista número 8, apresenta uma boa adaptação do jogo inglês, demonstrando que pelo menos a empresa em questão investiu tempo no estudo do software.

Quanto às suas dúvidas no programa Mazogs, se você já está com o tesouro, você só pode pegar uma espada se trocá-la por ele, ou seja, deve deixá-lo no lugar da espada, lutar com o monstro, arranjar outra espada e trocá-la novamente com o tesouro.

*No Simulador de Vôo, por o avião em **stall** significa fazê-lo perder a sustentação por diminuir excessivamente a velocidade, ou se, por acaso, o motor encrencar, ocorrendo então perda da sustentação (a única forma do avião encrencar no Simulador de Vôo é o término do combustível). Para evitar este tipo de acidente, você deve manter a velocidade compatível (acima de 90 km por hora) com a sustentação, tomando cuidado também com o combustível.*

Linha REM . . . revisitada

Tenho comprado a revista Microhoby desde o seu início e costumo rodar todos os programas que são apresentados e não tenho encontrado muitos problemas para digitá-los, a não ser com "A Linha REM", apresentada na Microhobby nº 7.

Estou precisando de uma linha REM com 3.000 pontos e tentei rodar o programa sem conseguir resultado. A cada tentativa, o programa cria uma linha REM com espaços em branco e repete a linha 3 REM duas ou mais vezes, distorcida e com palavras pela metade. Digitei o programa por diversas vezes e o resultado foi o mesmo.

Conferi o programa digitado a cada tentativa e estava de acordo com o publicado.

Francisco Antonio Kaupa
São Paulo — SP

Caro Francisco,

Ao invés de tentarmos corrigir o programa publicado na revista 7, faremos melhor, vamos publicar aqui uma versão mais simples do mesmo programa, para o TK85 de 48K.

```
   1 REM ..
   3 REM
   4 PRINT "COMPRIMENTO DA LINHA
REM"
   5 INPUT C
   6 PRINT C
   7 PAUSE 60
   8 POKE 16514,C-256*INT (C/256
)
   9 POKE 16515,INT (C/256)
  10 RAND USR 8192
  11 CLS
  12 LIST 2
```

A linha 1 REM contém apenas dois pontos. Eles conterão o comprimento da linha 2 REM, que será criada após você fornecer os dados necessários ao computador. As linhas 8 e 9 se encarregarão de colocar este valor no endereço correto. A linha 10 chama uma sub-rotina da ROM que servirá para criar a linha 2 REM com o número de pontos indicado pela linha 1 REM.

Após ter digitado todo o programa rode-o com o número de pontos que desejar, por exemplo, 3000. NUNCA digite zero ou um número de pontos superior à memória disponível em seu computador, pois isto causará CRASH!

Elimine as linhas desnecessárias e digite o programa que você deseja e pronto!

Mais um aviso: se você rodar este programa pela segunda vez, aparecerá outra linha 2 REM e mais um cursor. Por isso, antes de rodá-lo pela segunda vez, elimine a linha criada anteriormente. Se você precisar de mais de uma linha REM (vide Barão Vermelho II, publicado na revista 13), edite a linha 2 (naturalmente antes de introduzir qualquer código), alterando seu número para um maior do que 12. Após obter as linhas REM necessárias, elimine as linhas do programa e edite-as de acordo com seu programa. Se por acaso uma delas deva ter como número de linha zero, coloque 1 como número desta linha e dê:

POKE 16510,0

GRAVADOR

Tenho um TK 85. Gostaria que me ajudassem informando:

a) Qual o gravador que dá o melhor resultado no trato com o micro? Tenho um National meio sofisticado, mas que:

1) Não grava coisa alguma em "hi-save"; em qualquer volume, o VU meter atinge o nível máximo e quando tento o "hiverify", se consigo algum resultado, este tem sido sempre a indicação de erro.

2) Obriga-me, nos programas que colhem dados em fitas à parte, a diversas tentativas até que os dados entrem (. . .).

Eduvaldo Barbosa
Rio de Janeiro — RJ

Prezado Eduvaldo,

A rigor, não existe qualquer tipo de gravador que dê menos problemas, no que se refere à gravação/leitura de dados.

Você pode encontrar uma explicação detalhada nos artigos "Resolvendo os velhos problemas com o gravador", publicado nas Microhobby 12 e 13 e "A Gravação dos Programas", publicado na revista 2 e republicado na revista 14.

Estes artigos recomendam uma série de cuidados e fornecem alguns macetes para gravação e recuperação de programas. Entre estes macetes, um em particular merece um cuidado especial no seu caso: o ajuste do cabeçote. Qualquer tentativa de regular o cabeçote de um gravador mais sofisticado (um tape deck, por exemplo), pode redundar em fracasso e tornar o gravador incapaz de atingir a mesma performance anterior para outro tipo de aplicação (música, por exemplo). E, se isso ocorre, você vai precisar de um técnico realmente competente para por seu gravador novamente nos eixos. É recomendável você comprar um gravador de qualidade razoável (pode ser um portátil) e usá-lo para programas apenas.

Disco Voador e TK 2000

Inicialmente gostaria de congratulá-los pelos ótimos temperos dessa deliciosa revista que é a Microhobby.

Desejaria saber, também, se é possível para nós leitores, alterar alguns programas publicados na seção "Por Dentro do Apple", de forma que possam ser rodados num TK 2000, pois segundo o manual do TK 2000 existem basicamente três comandos (IN≠; PR≠; FLASH) que não são executados num TK 2000 mas o são num Apple. Todavia num programa recente desta seção (Disco Voador-publicado na revista 13), encontrei dois comandos (SCRN, na linha 280 e POP, na linha 640) que não existem no 2000.

Edgard Veiga Lion Neto
São Paulo — SP

Problemas com o INKEY$

Sou proprietário de um TK 2000 e já adquiri certa prática no seu manuseio. Porém, uma dúvida me impede de desenvolver diversos problemas. É o INKEY$.

Na seção "Desgrilando" da revista 13 foi apresentada uma rotina para substituir esta função. A minha intensão é fazer um programa que escreva na tela o número 1 quando eu digitar a tecla 1 e quando nada é pressionado, apareça o dígito Ø.

Veja o programa:

```
10 DATA 32, 67, 240, 133, 58, 96:
FOR F = 768 TO 773: READ X: POKE
F,X: NEXT
20 CALL 768
30 X = PEEK (58)
40 IF X = 1 THEN PRINT "1";
50 IF X = 0 THEN PRINT "0";
60 GOTO 20
```

Entretanto, por mais que eu digitasse o 1, só aparecia o zero.

Qual a forma correta do programa? Informo que, antes de digitar o programa, carreguei o Assembly em 0300.

Eduardo Camargo
São Paulo — SP

Prezado Eduardo,

Em primeiro lugar, não é necessário carregar o Assembly para carregar este programa em linguagem de máquina, porque isto é feito pela linha 10 automaticamente: os códigos estão na instrução DATA e são retirados pela instrução READ e colocados nas posições adequadas pelo POKE, enquanto é executado a malha FOR/NEXT.

Em segundo lugar, o código da tecla 1 é diferente do valor numérico 1. Assim, você deve trocar a linha 40 por:

```
40 IF X = 177 THEN PRINT "1";
```

Ao invés de usar esta rotina, você pode também usar o PEEK (39). Assim, para seu programa funcionar, retire a linha 10 e 20 e substitua as linhas 30, 40, 50, 60 por:

```
30 X = PEEK (39)
40 IF X = 23 THEN PRINT "1";
50 IF X = 48 PRINT "0";
60 GOTO 30
```

Para descobrir o código das outras teclas, digite o seguinte programinha:

```
10 PRINT PEEK (39): GOTO 10
```

Rode-o e anote um por um o valor das teclas pressionadas, obtendo assim uma tabela completa.

Caro Edgard,

No que se refere à tradução de programas, estamos procurando, sempre que possível, inserir comentários sobre modificações de textos ou listagens que se fazem necessárias para que os programas rodem num e noutro computador.

Contudo, há um pequeno equívoco: as instruções que você encontrou no disco voador, SCRN e POP, existem no TK 2000.

A instrução SCRN (X,Y) fornece o código da cor das coordenadas X e Y plotadas e POP tem um efeito semelhante ao RETURN, entretanto não executa o retorno ao programa principal.

O programa, ao invés de retornar ao programa principal, uma instrução após o último GOSUB, irá pular para o próximo GOSUB.

Quanto a outras modificações necessárias, estamos estudando para uma futura publicação a versão TK 2000 do programa Disco Voador.

Música por software para o TK 85

Gustavo Egídio de Almeida

Com este programa você poderá criar e armazenar músicas em seu computador de uma maneira mais completa. Experimente!

Como você deve ter notado, neste mês estamos apresentando uma série de programas e artigos sobre música. Coincidência ou não, isto está ocorrendo numa época em que a música está tomando conta do país, principalmente o *rock* — palavra até mal empregada por muitos conjuntos que proliferam por aqui, mas não correspondem a este estilo.

O problema é que este tipo de música está sendo divulgado em grande escala pelos meios de comunicação, principalmente as emissoras de FM, e as gravadoras multinacionais nos abarrotam de músicas com melodias medíocres e letras piores ainda, que não contém absolutamente nada de novo ou aproveitável, tornando-se totalmente descartáveis.

Porém, nós sentimos no ar que algo está mudando e, aos poucos, o *rock* verdadeiro vai penetrando em nossas vidas através de jornais, revistas, rádio e TV, de maneira mais sólida, provando que veio para ficar, deixando de lado o que uma sociedade consumista sempre ouviu, através de esquemas impostos pelas gravadoras multinacionais, que pensam ser donas da verdade e que, assim, estão "preservando a moral e os bons costumes".

Aos poucos porém, o verdadeiro *rock* vem chegando com toda a sua força e energia, derrubando essas barreira que, com o passar do tempo vão perdendo a sua rigidez, nos mostrando o que realmente interessa, ampliando a nossa consciência através de seus temas concretos, mostrando sua mensagem de paz e de esperança para a humanidade.

Conceitos básicos

Eis aqui um programa onde você poderá tocar melodias inteiras, dispondo de até duas oitavas completas com seus respectivos sustenidos e bemois.

Para entender melhor a linguagem que vamos falar, daremos aqui alguns conhecimentos básicos (figura 1):

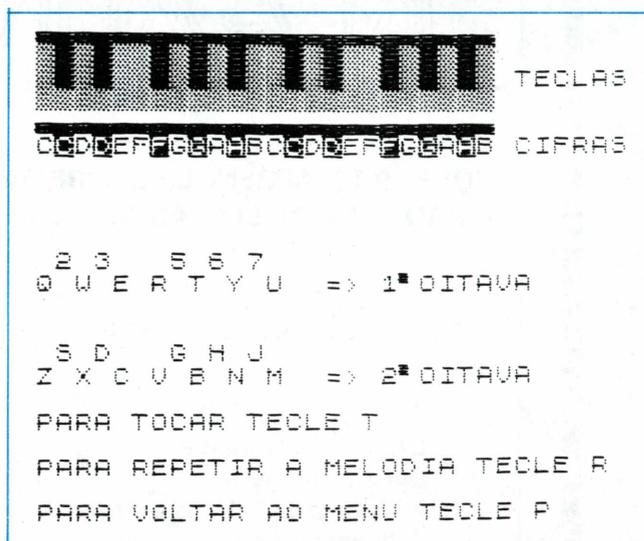



Figura 1

As notas musicais que compõem uma oitava são: DO, RÉ, MI, FA, SOL, LA, SI e DO. Elas podem ser representadas por letras, denominadas cifras:

DO	RE	MI	FA	SOL	LA	SI
C	D	E	F	G	A	B

As notas musicais são separadas entre si por tons (distância que separa uma nota de outra). Entre duas notas podem ocorrer outras, denominadas sustenidos ou bemois. Sustenido é a elevação de uma nota em meio tom e bemol é a degradação de uma nota em meio tom (figura 2).

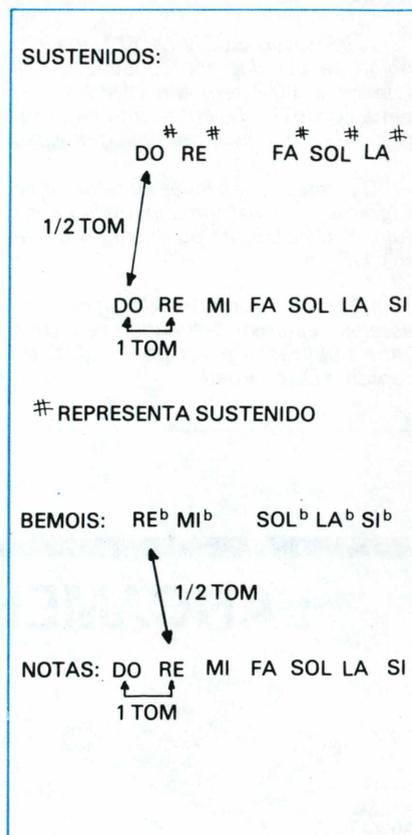



Figura 2

Como você deve ter notado, as notas SI e MI não apresentam sustenidos, nem as notas DO e FA apresentam bemois.

O programa

Agora, após essa rápida introdução sobre música, vamos ao que interessa: o programa propriamente dito.

Primeiro você deve digitar uma linha 1 REM com 76 caracteres. A seguir você deve digitar o programa da figura 3. Este programa permitirá a você introduzir os códigos decimais da figura 4.

Digite agora a listagem a seguir.

O programa é auto-explicativo, não exigindo maiores esclarecimentos. Após digitá-lo, grave-o através de GOTO 650.

Para ouvir os sons criados pelo programa, aumente o volume da TV ou conecte um amplificador na saída MIC do computador . . . e bons sons.

```
  1 REM ...........................
...................................
.....................
  10 FOR I=16514 TO 16590
  15 SCROLL
  20 PRINT I;
  30 INPUT N
  40 POKE I,N
  50 PRINT PEEK I
  60 NEXT I
```

Figura 3

16514	54
16515	0
16516	6
16517	255
16518	94
16519	175
16520	187
16521	40
16522	14
16523	219
16524	254
16525	123
16526	61
16527	32
16528	253
16529	211
16530	255
16531	123
16532	61
16533	32
16534	253
16535	24
16536	5
16537	14
16538	7
16539	13
16540	32
16541	253
16542	16
16543	231
16544	201
16545	0
16546	33
16547	48
16548	117
16549	205
16550	132
16551	64
16552	35
16553	126
16554	254
16555	255
16556	32
16557	247
16558	201
16559	33
16560	48
16561	117
16562	124
16563	50
16564	32
16565	78
16566	125
16567	50
16568	33
16569	78
16570	201
16571	58
16572	32
16573	78
16574	103
16575	58
16576	33
16577	78
16578	111
16579	205
16580	132
16581	64
16582	58
16583	33
16584	78
16585	60
16586	50
16587	33
16588	78
16589	201
16590	118

Figura 4

```
   1 REM Q ▗▘ COPY ?▐▌C:<= RETURN
 ?X4 CLEAR PEEK COPY ?X4 CLEAR /
▙:▛$4 CLEAR ( SCROLL TAN  5K?LN
▜RND75K??M4??M5?TAN U4??U5??LN ▜
RNDU5?WM5?TAN
   5 SLOW
  10 PRINT TAB 5;"MUSICA POR SOF
TWARE"
  20 PRINT AT 7,0;"COM ESTE PROG
RAMA, VOCE PODE      EXECUTAR MELO
DIAS DISPONDO DE    ATE DUAS OITA
VAS COMPLETAS."
  30 PRINT AT 21,5;"< PRESSIONE
ENTER >"
  40 PRINT AT 21,5;"> PRESSIONE
ENTER <"
  50 IF INKEY$="" THEN GOTO 30
  60 CLS
  65 PRINT "▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄
▀▀▀▀"
  70 FOR F=0 TO 1
  80 PRINT "▒█▒█▒█▒█▒█▒█▒█▒█▒█▒█▒█▒█▒
█▒█▒"
  90 NEXT F
 100 PRINT "▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒"
 105 PRINT "▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄
▀▀▀▀"
 110 FOR F=0 TO 1
 120 PRINT "C▣D▣EF▣G▣A▣B";
 130 NEXT F
 140 PRINT AT 2,25;"TECLAS"
 150 PRINT AT 5,25;"CIFRAS"
 160 PRINT AT 10,0;" 2 3   5 6 7
"
 170 PRINT "Q W E R T Y U  => 1ª
OITAVA"
 180 PRINT ,,,," S D   G H J"
 190 PRINT "Z X C V B N M  => 2ª
OITAVA"
 200 PRINT AT 17,0;"PARA TOCAR T
ECLE T"
 210 PRINT ,,"PARA REPETIR A MEL
ODIA TECLE R"
 215 PRINT ,,"PARA VOLTAR AO MEN
U TECLE P"
 220 IF INKEY$="T" THEN GOTO 300
 230 IF INKEY$="R" THEN GOTO 250
 240 GOTO 220
 250 CLS
 260 FAST
 270 RAND USR 16546
 275 SLOW
 280 GOTO 60
 300 CLS
 305 RAND USR 16559
 310 FAST
 320 LET HL=30000
 325 IF INKEY$="" THEN GOTO 325
 327 IF INKEY$="P" THEN GOTO 620
 330 IF INKEY$="Q" THEN POKE HL,
240
 340 IF INKEY$="2" THEN POKE HL,
229
 350 IF INKEY$="W" THEN POKE HL,
216
 360 IF INKEY$="3" THEN POKE HL,
207
 370 IF INKEY$="E" THEN POKE HL,
195
 380 IF INKEY$="R" THEN POKE HL,
184
 390 IF INKEY$="5" THEN POKE HL,
173
 400 IF INKEY$="T" THEN POKE HL,
165
 410 IF INKEY$="6" THEN POKE HL,
156
 420 IF INKEY$="Y" THEN POKE HL,
148
 430 IF INKEY$="7" THEN POKE HL,
139
 440 IF INKEY$="U" THEN POKE HL,
131
 450 IF INKEY$="Z" THEN POKE HL,
122
 460 IF INKEY$="S" THEN POKE HL,
115
 470 IF INKEY$="X" THEN POKE HL,
107
 480 IF INKEY$="D" THEN POKE HL,
103
 490 IF INKEY$="C" THEN POKE HL,
96
 500 IF INKEY$="V" THEN POKE HL,
90
 510 IF INKEY$="G" THEN POKE HL,
86
 520 IF INKEY$="B" THEN POKE HL,
81
 530 IF INKEY$="H" THEN POKE HL,
75
 540 IF INKEY$="N" THEN POKE HL,
72
 550 IF INKEY$="J" THEN POKE HL,
68
 560 IF INKEY$="M" THEN POKE HL,
40
 590 RAND USR 16571
 600 LET HL=HL+1
 610 GOTO 325
 620 SLOW
 630 POKE HL,255
 640 GOTO 60
 650 SAVE "MUSIC▣"
 660 SLOW
 670 RUN
```

HOBBYSHOP VEJA SE SUA CIDADE TEM O QUE VOCÊ PRECISA

SÃO PAULO

CURSO
ASSEMBLER Z-80 PARA SINCLAIR
com BERNHARD W. SCHÖN

O curso começa dia 11.03.85 no **SENAC INFORMÁTICA.** As inscrições já estão abertas. Rua Dr. Vila Nova, 228 — Tel.: **255-0066.**

ASPP

Fundamentum sistemas Lógicos

Cibernética, informática & Microcomputadores

Cursos para crianças, jovens e adultos
Basic I e Basic avançado
Rua Wanderley, 480 — Perdizes — Fone: 62-5385.

S.O.S COMPUTADORES

Cursos Basic, Cobol, Assembler
A nova maneira de aprender a programar.
Núcleo I: Av Pacaembú, 1280 — Fone: 66-7656 — *SP.*

ENG Comércio de Computadores Ltda.
TK85 x TK2000?
Só na ENG você adquiri o seu TK2000 nas melhores condições e ainda dá o seu velho TK83, TK85 ou CP200 como parte de pagamento. TK2000 é na ENG. Showroom — Tel. 284-1218. Av. Paulista, 1159 — Cj. 611.

---

micro-total sistemas

CURSOS DE BASIC, PLANILHA ELETRÔNICA, PROCESSAMENTO DE TEXTOS E OUTROS

Método moderno, todo apostilado com grande enfoque em sistemas.
Consulte-nos hoje mesmo.
Rua Pinheiros, 1361 — Fone: 813-8585 — Pinheiros — São Paulo

MONOLITH 2001
Eletrônica e Jogos Com. e Exp. Ltda.
Equipamentos TK85, Elppa II, Elppa Jr. e Color 64. Jogos em Geral.
Rua Augusta, 1371 S/L7 — Fone: 268-4370 — *SP.*

SÃO PAULO

DATA RECORD INFORMÁTICA
Cursos, Consultoria, Ass. Técnica e Suprimentos
Comece 85 programando!
- Cursos Intensivos
- Reserva de vagas

Cursos: Cobol • Basic I e II • Debase II • Aplicativos
Vantagens: 1 aluno p/ Micro • Professores da USP • Estágio garantido • Bolsas de Estudo
Av. Sto. Amaro, 5450 — Brooklin — Tel.: 543-9937.

S. BERNARDO DO CAMPO — SP

MICROCOMPUTADOR JÁ

POLIDATA

Software-House especializada no desenvolvimento de sistemas e cursos de treinamento para microcomputadores.

BASIC COBOL CP/M VISICALC DBASE II E OUTROS

Filial: R. Domingos J. Ballotim, 46 - 5º cj. 55 - CEP 09700 - S.B. do Campo - Tel. 448-5970

ABC COMPUTAÇÃO A POLIVALENTE DA INFORMÁTICA:

Cursos Basic, Assembler e Cobol. Microcomputadores — Suprimentos, Calculadoras, Órgãos Eletrônicos, Software, Microclub.
Av. Senador Vergueiro, 4962 — 1º andar — Sala 6 — Rudge Ramos — S. B. Campo — CEP 09720 — Tel. 455-1940.

BAHIA

Sua empresa poderia estar aqui.
Anuncie no HOBBYSHOP e todos os Leitores da região conhecerão sua empresa.
Anúncio econônico e de retorno garantido.

RIO DE JANEIRO

PROSERV-Processamento Dados.Cursos e Rep.Ltda.
.MICROCOMPUTADORES (Novos e Usados)
.CURSOS (Cobol. Basic. CP/M. DBase II)
.SUPRIMENTOS (Formularios. Disquetes. Fitas. etc.)
.LIVROS E REVISTAS
.SOFTWARE (TRS80. Apple. TK85)
Lg.Nove de Abril 27 salas 626/628
Tel: (0243) 429800 - V.Redonda - RJ

MINAS GERAIS

MICRO E VIDEO

Curso de Basic com turmas mensais
Programas para toda linha de microcomputadores — Sinclair, TRS-80, Apple, TRS Color, Comodore CP/M — Aplicativos e Jogos (Solicite catálogo especificando seu equipamento).
Livros e revistas nacionais e estrangeiros. Venda de Micros, periféricos e suprimentos. Soft House.
VILLABELLA SHOPPING — LOJA 6
Avenida Japão, 229 — Cariru — CEP 35160 — Fone (031) 821-2888 — Ipatinga — MG.

TK WORD

TK-Word, processador de texto para o TK

O produto que enfocaremos este mês na seção "Analisando" é um processador de textos para computadores compatíveis com o TK 83 com expansão, ou TK 85 de 16 ou 48 k, fabricado pela Cibertron.

Um processador de texto desenvolvido para o TK 85 demonstra, mais uma vez, que computadores do seu porte não são simples brinquedos, mas servem para aplicações sérias, como é este aplicativo.

Teste e texto final: Alvaro A. L. Domingues

Escrever qualquer coisa implica num processo interativo em que rabiscamos, alteramos, apagamos, rasgamos inúmeras folhas de papel até termos algo que nos agrade e que seja isento de erros.

E se você pudesse escrever só a versão definitiva daquilo que você está escrevendo? Impossível? Não se você tiver um computador e uma impressora. O que transforma seu computador numa máquina de escrever com inúmeros recursos é um aplicativo chamado processador de textos. Se você possui um computador compatível com o Apple, dispõe de várias versões do Magic Window; se tiver um compatível com o TRS-80, uma versão do Scripsit ou do Super Scripsit. E se possuir um TK 83 ou 85?

A Cibertron, empresa paulista de software para o TK 83 e 85 e seus compatíveis, nos enviou, para análise, um processador de textos para esta linha de computadores.

Apresentação

A fita vem numa embalagem bastante simples, onde constam o nome da empresa, do produto, os computadores que rodam o programa e a memória requerida (16 k).

A fita é de boa qualidade e, além disso, os fabricantes gravaram o programa em ambos os lados, visando uma garantia de que se uma versão não entrar, existe outra que pode cobrir a falha.

Acompanha um manual em folhas de xerox, muito bem explicado, não deixando o usuário "na mão" em nenhum momento. Encontramos um erro de datilografia que pode confundir o leitor: a palavra RUBOUT aparece associada a duas funções diferentes, mas no MENU de funções do programa este item aparece correto.

Carregamento e teste do programa

Não tivemos problemas com o carregamento do programa, apesar da sua extensão, revelando tratar-se de uma boa gravação. O programa entra "rodando", mostrando uma mensagem inicial. Pouco depois aparece a "janela" de 22 linhas, onde poderão ser escritos os textos. O arquivo de texto possui um tamanho máximo de 300 linhas e mínimo de 22.

Testamos o programa digitando o texto da figura 1. Durante a digitação tivemos a oportunidade de usar a maioria dos comandos e observar o comportamento do programa.

Em primeiro lugar, aparece um cursor retângular que acompanha a digitação. Quando uma palavra ultrapassa os limites da tela, o cursor muda para quadrangular indicando ao usuário que ocorreu estouro da linha. Se o usuário digitar mais algumas letra, a palavra inteira é transferida para a outra linha e a linha que estava sendo digitada é preenchida com espaços entre as palavras, igualmente distribuídos, blocando a linha, ou seja, alinhado-a pelas margens ou parágrafos, como mostramos nos textos das figuras 1 e 2.

```
TEXTO    PARA    VERIFICACAO    DO
            TK-WORD

  VERIFICAREMOS ALGUNS COMANDOS
ESPECIAIS.

  AS     TECLAS     DESTINADAS     A
DETERMINADAS      FUNCOES      E
INSTRUCOES PERMITEM  REALIZAR-SE
OUTRAS FUNCOES,  ESPECIFICAS  DO
PROCESSAMENTO  DE  TEXTOS  COMO,
POR EXEMPLO:

LLIST MOSTRA O GLOSSARIO;
<> CENTRALIZA;
AND  PERMITE   A   INCLUSAO   DE
CARACTERES; ETC..

  O CAMPO DESTINADO A  TEXTO  E*
DE 300 LINHAS.

  QUANDO A TELA E* CHEIA,  OCORRE
UM  SCROLL  AUTOMA*TICO  PARA  A
LINHA  SEGUINTE.  PARA  PODERMOS
VER   AS    LINHAS    DO    ARQUIVO,
DEVEMOS USAR O COMANDO THEN, QUE
DA* UM SCROLL, DESCENDO O TEXTO.

           FIM DO TEXTO
```

Texto para teste do TK-Word **Figura 1**

Neste exemplo testamos todos os comandos e verificamos o comportamento do "scrolling", da transferência, da inclusão e retirada de caracteres e linhas.

```
TEXTO PARA VERIFICACAO DO
TK-WORD

VERIFICAREMOS ALGUNS COMANDOS
ESPECIAIS.

AS TECLAS DESTINADAS A
DETERMINADAS FUNCOES E
INSTRUCOES PERMITEM REALIZAR-SE
OUTRAS FUNCOES, ESPECIFICAS DO
PROCESSAMENTO DE TEXTOS COMO,
POR EXEMPLO:

LLIST MOSTRA O GLOSSARIO;
<> CENTRALIZA;
AND PERMITE A INCLUSAO DE
CARACTERES; ETC..

O CAMPO DESTINADO A TEXTO E'
DE 300 LINHAS.

QUANDO A TELA E' CHEIA, OCORRE
UM SCROLL AUTOMA'TICO PARA A
LINHA SEGUINTE. PARA PODERMOS
VER AS LINHAS DO ARQUIVO,
DEVEMOS USAR O COMANDO THEN, QUE
DA' UM SCROLL, DESCENDO O TEXTO.

FIM DO TEXTO
```

Figura 2

Centralizando.
O mesmo texto, agora centralizado, mostrando uma outra forma de apresentação.

Durante a digitação pudemos usar vários comandos obtidos pela digitação de SHIFT + tecla.

Estes comandos são redefinições de instruções e comandos BASIC. Algumas destas funções têm alguma associação com o seu novo significado (por exemplo AND para inclusão de linhas e caracteres) outras não; mas isto é compreensível devido à própria estrutura do programa e do teclado do TK 83/85.

TABELA I — Comandos do TK-Word

Comando Original	Nova função
EDIT	apaga linha
AND	insere linha ou caractere
THEN	SCROLL para cima
TO	SCROLL para baixo
GRAPHICS (SHIFT 9)	Inverte/Desinverte caractere
RUBOUT	Apaga caractere
"GRAPHICS" (SHIFT Q)	"Desacerta" linha
OR	LIG/DES "Transferência"
STEP	Reacerta parágrafo
< >	Centraliza linha
≠	Desloca frase para direita
≠	Desloca frase para esquerda
STOP	Mostra menu com as seguintes opções: grava, programa com arquivo, limpa, grava programa sem arquivo ou pára o processamento
LPRINT	Imprime arquivo
SLOW	Mostra o início do arquivo
FAST	Mostra o fim do arquivo
LLIST	Mostra este menu
NEW LINE	Vai para a linha seguinte
Setas	movimentam o cursor
**	Acerta linha

Recursos

O TK-Word, apesar de ter sido desenvolvido para computadores pessoais de pequeno porte, tem a maioria dos recursos de um processador de texto de computadores maiores (tabela I). Tivemos alguma dificuldade, contornável, em corrigir linhas que haviam sido divididas e blocadas, quando se suprimia letras ou palavras. Um pouco de treino e conhecimento do programa permite ao usuário descobrir todas as técnicas para tirar o máximo proveito do aplicativo.

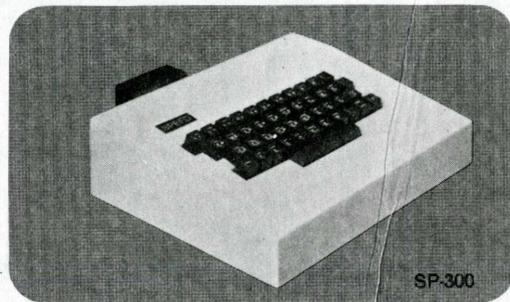
O texto pode ser gravado juntamente com o programa, proporcionando ao usuário a possibilidade de arquivar vários textos para uso futuro ou, até, como uma forma original de "fono-postal", caso não possua impressora e queira se comunicar com outro possuidor do TK.

O Programa

O programa possui várias rotinas em linguagem de máquina, bastante extensa, o que mostra que seu autor possui um grande conhecimento da máquina a nível de Assembly. Estas rotinas abrigam a parte principal do programa; em BASIC estão apenas algumas rotinas e o gerenciamento total do programa.

A parte em BASIC está bem estruturada, sem o uso de memória desnecessária.

Conclusão

O programa TK-Word explora ao máximo a capacidade do TK 83/85 para o tratamento de texto. Com o uso de uma impressora de boa qualidade, ou uma interface para impressora paralela, pode-se conseguir um sistema de processamento de textos barato e relativamente eficiente.

Números Complexos

Daniel R. Falconer
Wilson José Tucci

Poucas linguagens de programação permitem ao programador trabalhar diretamente com valores complexos. Dentre as que permitem destacam-se o FORTRAN e o PL/I. Embora o Pascal não tenha o tipo de dado complexo pré-definido, é muito fácil implementar para complexos as operações existentes para números reais.

Um número complexo Z pode ser representado como par ordenado ou como a soma de dois valores, um real e outro imaginário, assim:

$$Z = (a.b) = \underbrace{a}_{\text{real}} + \underbrace{ib}_{\text{imaginário}}$$

a é chamado de parte real de Z e *b* é chamado de parte imaginária de Z.

A própria estrutura dessa representação algébrica dos números complexos nos leva a definir o tipo COMPLEX como um registro de dois valores reais:

```
TYPE COMPLEX = RECORD

    RE, IM : REAL
END;
```

Para implementar as operações não vamos fazer um programa, mas um UNIT, que será instalado na biblioteca do PASCAL, para depois ser usado por qualquer programa.

Para cada uma das operações básicas, faremos um procedimento que recebe 2 parâmetros complexos de entrada e devolve um complexo que é o resultado da operação:

operação	procedimento
+	C SOM
—	C SUB
*	C MUL
/	C DIV

Para escrever os procedimentos, usamos as seguintes relações:

$$(a+ib) + (c+id) = (a+c) + i(b+d)$$
$$(a+ib) - (c+id) = (a-c) + i(b-d)$$
$$(a+ib) * (c+id) = (ac-bd) + i(bc + ad)$$

$$\frac{(a+ib)}{(c+id)} = \frac{(a+ib) . (c-id)}{c^2 + d^2}$$

Depois que você digitar e compilar o UNIT fornecido, basta instalá-lo na SYSTEM LIBRARY, conforme descrito nas páginas 186-193 do Apple Pascal Operating System Reference Manual.

A seguir apresentamos a Listagem completa para tal implementação, tirada na Mônica da Elebra Informática.

```
      (*$S+*)
UNIT COMOP; INTRINSIC CODE 25 DATA 26;

  INTERFACE

    TYPE COMPLEX = RECORD
                    RE, IM : REAL;
                  END;
    VAR Z : COMPLEX;
    PROCEDURE CSUM(X,Y: COMPLEX; VAR Z:COMPLEX);
    PROCEDURE CSUB(X,Y: COMPLEX; VAR Z:COMPLEX);
    PROCEDURE CMUL(X,Y: COMPLEX; VAR Z:COMPLEX);
    PROCEDURE CDIV(X,Y: COMPLEX; VAR Z:COMPLEX);


  IMPLEMENTATION

    PROCEDURE CSUM;
      BEGIN
        Z.RE := X.RE + Y.RE;
        Z.IM := X.IM + Y.IM;
      END;

    PROCEDURE CSUB;
      BEGIN
        Z.RE := X.RE - Y.RE;
        Z.IM := X.IM - Y.IM;
      END;

    PROCEDURE CMUL;
      BEGIN
        Z.RE := X.RE*Y.RE - X.IM*Y.IM;
        Z.IM := X.IM*Y.RE + X.RE*Y.IM;
      END;

    PROCEDURE CDIV;
      VAR D : REAL;
      BEGIN
        Y.IM := - Y.IM;
        CMUL(X,Y,Z);
        D := (Y.RE*Y.RE+Y.IM*Y.IM);
        Z.RE := Z.RE/D;
        Z.IM := Z.IM/D;
      END;

    BEGIN
    END.
```

Algumas Rotinas para Matrizes

Calculadoras sofisticadas como a HP-41, HP-15, TI-58 e TI-59, freqüentemente possuem um pacote para operações com matrizes, geralmente incluindo cálculo de determinantes, cálculo da matriz inversa e resolução de sistemas de equações. Operações com matrizes geralmente envolvem tantas contas, que essas calculadoras podem levar vários minutos para inverter uma matriz 10*10.

Por que não usar a velocidade bem maior do microcomputador quando este estiver disponível? É para isso que apresentamos algumas rotinas para manipular matrizes, que podem servir de bases para seus próprios programas de matrizes. Para que você possa experimentá-los e usá-los para alguns fins específicos, essas rotinas estão agrupadas em um programa que:

- triangulariza matrizes
- calcula determinantes
- resolve sistemas de equações lineares

A seguir descrevemos o programa e, junto com ele, a teoria de matrizes que foi usada para desenvolvê-lo.

Linha 20 — 140: esse é o "programa principal". Na verdade funciona como um menu para escolher as diversas operações possíveis que são:

1 — entrada da matriz A
2 — edição da matriz A
3 — triangularização da matriz A
4 — resolução de sistema
5 — cálculo do determinante da matriz A
6 — fim

O menu só aceita números válidos (veja as linhas 110 e 120).

Linhas 1000-1130: essa rotina faz a entrada da matriz A; perceba que primeiro é obtido o seu tamanho e só então ela e outros valores são dimensionados. O CLEAR é necessário para apagar a matriz A da última vez que essa rotina foi utilizada. Como o CLEAR perturba a pilha de endereços da sub-rotina, é necessário voltar para um GOTO. A parte realmente interessante são as linhas 1070 — 1120.

Linhas 2000 — 2150: permitem ver e modificar os elementos da matriz A depois que ela já foi digitada (é comum errar-se pela menos um elemento ao se digitar uma matriz grande), a matriz é mostrada coluna a coluna; para se modificar um elemento de uma coluna, basta dar o número da sua linha (o que aparece na frente do valor) e dar RETURN. O computador pede o novo valor. Para passar para a próxima coluna ou acabar (se você estiver na última coluna), digite 0 e RETURN.

Essa rotina também pode ser usada para ver a matriz A após alguma operação, como a triangularização, por exemplo.

Linhas 3000 — 3200: faz a triangularização da matriz. A triangularização é uma operação fundamental para se calcular determinante ou resolver sistema de equações. O objetivo da triangularização é deixar todos os elementos de um dos lados da diagonal principal com valor 0. Essa rotina zera o lado de baixo da diagonal inferior, através das seguintes operações: para as linhas de número 1 até N—1 substitua as linhas abaixo de 1 pela própria linha menos a linha 1, vezes um fator tal que zere o elemento da coluna 1 de cada uma dessas linhas.

Essa rotina só funcionará se todos os elementos da diagonal principal (com possível exceção do último), forem diferentes de zero. Se a rotina não funcionar é porque o determinante da matriz é 0 e o sistema é impossível ou indeterminado. Se você editar a matriz depois da triangularização, vai perceber que os elementos abaixo da diagonal principal são diferentes de 0, pois aí foram guardados os fatores multiplicativos. (Mas os elementos da matriz A abaixo da diagonal principal são 0).

Linhas 4000 — 4320: resolução de sistemas; depois que a matriz A estiver triangularizada, podemos resolver a equação Ax = b. As linhas 4020 — 4070 fazem a entrada do vetor de termos independentes b.

Linhas 4090 — 4140: repetem em b as operçaões feitas para triangularizar a matriz A.

Linhas 4160 — 4230: calculam o vetor das incógnitas x de baixo para cima, utilizando as fórmulas:

$$X_n = \frac{bn}{a_{nn}}$$

$$xi = \frac{bi - \sum_{j=i+1}^{n} aijxj}{a_{nn}}$$

Linhas 4250 — 4320: imprimem o vetor das incógnitas e esperam você apertar alguma tecla. Essa rotina só pode ser executada se a triangularização ocorreu e o determinante de A for diferente de 0. Verifique antes.

Linhas 5000 — 5090: calculam o determinante da matriz A depois que ela foi triangularizada; para isto basta multiplicar os elementos da diagonal principal.

A listagem desse artigo foi obtida na impressora Mônica da Elebra Informática.

```
 LIST

10  REM  UM PACOTE DE ROTINAS
      PARA MATRIZES
20  HOME
30  VTAB 3: INVERSE : PRINT "RESO
     LUCAO DE SISTEMAS LINEARES":
     NORMAL
40  VTAB 5: PRINT "1 - ENTRADA DA
     MATRIZ A"
50  PRINT "2 - EDICAO DA MATRIZ A
     "
60  PRINT "3 - TRIANGULARIZACAO D
     A MATRIZ A"
70  PRINT "4 - RESOLUCAO DE SISTE
     MA"
80  PRINT "5 - CALCULO DO DETERMI
     NANTE DA MATRIZ A"
90  PRINT "6 - FIM"
100  VTAB 15: HTAB 1: PRINT "QUAL
     A OPERACAO DESEJADA ? ";: GET
     A$:A =  VAL (A$)
110  IF A = 6 THEN  END
120  IF A < 1 OR A > 5 THEN 100
130  ON A GOSUB 1000,2000,3000,40
     00,5000
140  GOTO 10
1000  REM    DIMENSIONAMENTO E
1010  REM  ENTRADA DA MATRIZ A
1020  REM  LIMPAR MEMORIA
1030  CLEAR : HOME
1040  INPUT "QUAL A ORDEM DO SIST
     EMA ? ";N
1050  PRINT : PRINT
1060  DIM A(N,N),B(N),X(N)
1070  FOR I = 1 TO N
1080  FOR J = 1 TO N
1090  PRINT "A(";I;",";J;") ? ";
1100  INPUT "";A(I,J)
1110  NEXT J
1120  NEXT I
1130  GOTO 10
2000  REM  EDICAO DA MATRIZ A
2010  FOR J = 1 TO N
2020  HOME
2030  PRINT " COLUNA ";J
2040  PRINT : PRINT
2050  FOR I = 1 TO N
2060  PRINT I;" - ";A(I,J)
2070  NEXT I
2080  VTAB 23: PRINT "NUMERO A CO
     RRIGIR (0 PASSA )";
2090  INPUT L
2100  IF L = 0 GOTO 2140
2110  VTAB 23: HTAB 1: PRINT  SPC(
     39)
2120  VTAB 23: HTAB 1: PRINT "A("
     ;L;",";J;") ? ";: INPUT "";A
     (L,J)
2130  GOTO 2020
2140  NEXT J
2150  RETURN
3000  REM  TRIANGULARIZACAO DA
3010  REM  MATRIZ A
3020  FOR I = 1 TO N - 1
3030  FOR J = I + 1 TO N
3040  IF A(I,I) = 0 THEN 3150
3050 C = A(J,I) / A(I,I)
3060  FOR K = I + 1 TO N
3070 A(J,K) = A(J,K) - A(I,K) * C

3080  NEXT K
3090  REM  GUARDA O FATOR
3100  REM  MULTIPLICATIVO
3110 A(J,I) = C
3120  NEXT J
3130  NEXT I
3140  GOTO 3200
3150  PRINT : PRINT : PRINT "TRIA
     NGULARIZACAO INCOMPLETA"
3160  PRINT : PRINT "DET A = 0"
3170  POKE  - 16368,0
3180 KEY =  PEEK ( - 16384)
3190  IF KEY < 128 THEN 3180
3200  RETURN
4000  REM  RESOLUCAO DE SISTEMA
4010  REM  ENTRADA DO VETOR B
4020  HOME
4030  PRINT : PRINT
4040  FOR I = 1 TO N
4050  PRINT "B(";I;") ? ";
4060  INPUT "";B(I)
4070  NEXT I
4080  REM  OPERACOES EM B
4090  FOR I = 1 TO N - 1
4100  FOR J = I + 1 TO N
4110 C = A(J,I)
4120 B(J) = B(J) - B(I) * C
4130  NEXT J
4140  NEXT I
4150  REM  CALCULO DAS INCOGNITAS

4160 X(N) = B(N) / A(N,N)
4170  FOR I = N - 1 TO 1 STEP  -
     1
4180 SOMA = 0
4190  FOR J = I + 1 TO N
4200 SOMA = SOMA + X(J) * A(I,J)
4210  NEXT J
4220 X(I) = (B(I) - SOMA) / A(I,I
     )
```

```
4230  NEXT I
4240  REM  IMPRIME AS RAIZES
4250  PRINT
4260  FOR I = 1 TO N
4270  PRINT "X(";I;") = ";X(I)
4280  NEXT I
4290  POKE  - 16368,0
4300 KEY =  PEEK ( - 16384)
4310  IF KEY < 128 THEN 4300
4320  RETURN
5000  REM CALCULO DO DETERMINANTE

5010 DET = 1
5020  FOR I = 1 TO N
5030 DET = DET * A(I,I)
5040  NEXT I
5050  PRINT : PRINT : PRINT "DET
    DA MATRIZ A = ";DET
5060  POKE  - 16368,0
5070 KEY =  PEEK ( - 16384)
5080  IF KEY < 128 THEN 5070
5090  RETURN
```

TENTE ESTA

Estes são para o TK 83/85

Daniel Nordemann

Estes dois programas curtíssimos são para enfeitar a tela da sua televisão com quadros animados. Operar em Slow.

```
  1 PRINT AT 21*RND,31*RND;CHR$
(6*RND)
  2 RUN
```

e:

```
  1 PLOT 63*RND*RND,43*RND*RND
  2 UNPLOT 63*RND,43*RND
  3 RUN
```

São fáceis de entender e podem ser modificados à vontade (diminuir os quadros, usar menos caracteres, até um só se quiser, misturar os dois programas, alterar a densidade de caracteres.

O Apresentador
Memória ocupada: 6912 bytes
Soma Sintática: 28438

Nível: (2 computadores)

Computadores: TK-83 e compatíveis

O Apresentador

Bernardo Stein

Muito bem,

Passaram há muito as festas de fim de ano e você é um felizardo que ganhou um aparelho de video-cassete e uma câmera de vídeo no Natal.

Todas as suas horas vagas são agora passadas com o seu novo "brinquedo".

E o seu computador?

Há algum tempo atrás ele era o centro de suas atenções. Agora, ele está sendo colocado de lado e esquecido.

Que tal mudarmos este estado de coisas?

Vamos criar dois programas para o TK que irão permitir fazer espetaculares títulos e finais para as suas realizações em video-cassete.

Com isso você vai ter oportunidade de dedicar seu tempo livre aos dois equipamentos e ainda por cima, vai "profissionalizar" as suas obras.

Com o ***Apresentador*** você criará uma abertura para seus programas com até cinco telas diferentes, para os títulos e sub-títulos, que se sucedem uns aos outros na tela de sua TV.

Você vai precisar de algumas horas de paciência e de empenho para entrar com o programa no seu TK e verá que no fim terá valido a pena.

Se você só possui o aparelho de vídeo-cassete e ainda não a câmara, ainda assim o programa será de muita utilidade, pois você poderá fazer aberturas para programas gravados diretamente da televisão.

Introdução da rotina em código de máquina

a) Entre com o "programa inicial" mostrado na figura 1 (Atenção! A linha 1 REM O APRESENTADOR — (C) STEIN deve ser digitada exatamente como está, pois nela se alojará a rotina em código de máquina).

b) RUN o programa e introduza com cuidado um a um os valores indicados na "Listagem de códigos" da figura 2.

c) Após entrar com os valores, apague as linhas 10 e 60 (entre com o número da linha e NEW LINE) deixando somente a linha 1 REM que agora terá uma nova e estranha aparência.

Introdução do programa principal

Mantendo a linha 1 REM, entre todo o programa da listagem principal da figura 3.

Sugiro armazenar em fita, algumas vezes o programa durante a sua introdução, para não perder todo seu trabalho em caso de uma falha.

Uma vez completada a introdução, armazene o programa em fita entrando SAVE "O Apresentador" e "NEW LINE" (não entre GOTO 7060).

```
          PROGRAMA INICIAL

   1 REM O APRESENTADOR (C)STEIN
  10 FOR N=16514 TO 16536
  20 PRINT N;"...";
  30 INPUT P
  40 PRINT P,
  50 POKE N,P
  60 NEXT N
```

Fig. 1

```
       LISTAGEM DE CODIGOS
     PARA O PROGRAMA INICIAL

16514...237        16515...91
16516...123        16517...64
16518...42         16519...12
16520...64         16521...1
16522...215        16523...2
16524...237        16525...176
16526...201        16527...237
16528...91         16529...12
16530...64         16531...42
16532...123        16533...64
16534...24         16535...241
16536...201
```

Fig. 2

CRIAÇÃO DOS TÍTULOS

Suponhamos agora que o seu apelido familiar seja DINDO e que você seja um tio legal que se propõe a gravar em vídeo o oitavo aniversário de seu sobrinho MARCELO.

Proceda da seguinte forma:

1 — LOAD o Apresentador no computador e entre GOTO 2000 e NEW LINE. Aparecerá o menu abaixo:
O Apresentador
Você quer
1. Criar uma apresentação completa
2. Fazer / refazer só os títulos
3. Ver uma apresentação completa
4. Ver só os títulos
5. SAVE a apresentação sem/com os títulos
Escolha a opção 1.

2—O TK pedirá a você para criar a abertura com duas linhas, cada uma com até oito letras por linha.

No nosso exemplo (Figura 5) a primeira linha será DINDO com cinco letras; informe isso ao TK, ele então entrará no modo FAST e a imagem desaparecerá por alguns segundos.

Ao ser perguntado, informe ao TK que a segunda linha terá quatro letras (no nosso caso, "NEWS"). Completando a abertura o computador lhe mostrará como ela ficou. Se você não gostou, entre "N" e repita a abertura. Se gostou, entre "S" e o programa voltará ao menu.

3 — Neste ponto, se escolher a opção número 5 você poderá armazenar a abertura para usar em outras ocasiões. Ao dar LOAD ao programa, ele irá automaticamente na opção 2 na qual você irá criar os títulos.

4 — Ao invés de SAVE, você poderá escolher diretamente a opção número 2 e criar títulos que comporão a apresentação completa.

5 — Informe ao computador quando ele perguntar quantas telas comporão os títulos (no máximo cinco telas). No nosso exemplo são duas telas (Figura 6 e 7).

6 — O computador perguntará então quantas linhas terá a primeira tela.

7 — Agora informe que a primeira linha terá sete letras (MARCELO).

8 — O computador então entrará no modo FAST e ampliará esta primeira linha.

9 — Prossiga com as outras três linhas:
2ª linha — 3 letras — FAZ
3ª linha — 1 letra — 8
4ª linha — 4 letras — ANOS

10 — No fim o seu TK lhe apresentará a tela para julgamento.

11 — Entre "N" se quiser refazê-la.

12 — Entre "S" se estiver OK.

13 — Crie então a segunda tela (Figura 7):
1ª linha da 2ª tela — 1 letra — E
2ª linha da 2ª tela — 7 letras — CONVIDA
3ª linha da 2ª tela — 4 letras — TODA
4ª linha da 2ª tela — 6 letras — PATOTA

14 — Julgue a tela depois de concluída.

15 — Se você entrar "S" a apresentação toda estará concluída e você poderá admirá-la.

16 — Entre "M" depois de concluída a apresentação, para voltar ao MENU.

17 — Neste ponto, escolha a opção 5 e "SAVE" no gravador toda a apresentação.

18 — Se em algum momento da criação da apresentação o programa se interromper, entre GOTO 2000 e escolha a opção 1 ou 2.

GRAVAÇÃO DA APRESENTAÇÃO EM VÍDEO-CASSETE

Chegou o grande dia. É hoje a festa do Marcelo. Acorde cedo, faça sua ginástica para se sentir disposto. Depois monte o seu vídeo-cassete e o seu TK conforme o esquema da Figura 9. Interligue a saída do computador à entrada da antena do vídeo cassete.

Se o seu vídeo-cassete não tiver recurso de "AUDIO DUBBING" (que permite a gravação de trilha musical em separado) você deverá ligar uma fonte de som (microfone, gravador, toca-disco etc. . .) diretamente na entrada AUDIO IN (veja também na Fig. 9).

Carregue o programa "O Apresentador" no computador e coloque uma fita (ou disco) na fonte de som. Coloque uma fita de vídeo nova no aparelho de vídeo cassete.

ATENÇÃO: Ligue o vídeo cassete, sintonize o canal correspondente ao canal em que você opera o seu computador. REGULE a sintonia fina deste canal já que ele normalmente não vem regulado de fábrica.

Faça isto até obter uma imagem nítida do computador na tela de TV ligada ao vídeo-cassete.

Você poderá então, passar a frente. Coloque o vídeo-cassete em REC-PLAY e PAUSE; prepare a fonte de som, e deixe o TK no MENU. Já que você não tem três mãos, peça a ajuda de alguém para soltar o PAUSE do vídeo, escolher a opção nº 3 do MENU no TK e liberar a fonte de som.

```
O APRESENTADOR

VOCE QUER.

1. CRIAR UMA APRESENTACAO
2. FAZER/REFAZER OS TITULOS
3. VER UMA APRESENTACAO COMPLETA
4. VER SO OS TITULOS
5. "SAVE" A APRESENTACAO SEM/COM
   OS TITULOS
```

Fig. 4

```
DINDO
NEWS
apresenta
```

Fig. 5

```
MARCELO
FAZ
8
ANOS
```

Fig. 6

```
E
CONVIDA
TODA
PATOTA
```

Fig. 7

FIGURA 3

```
   1 REM GOSUB ??RNDE£RND" NOT "
GOSUB ?TAN GOSUB ?£RNDE?RND/ LET
 TAN
   2 FOR C=-1 TO 0
   3 SLOW
   4 CLS
   5 PRINT "1. APRESENTACAO",,,
   6 PRINT "QUANTAS LETRAS TERA
A ";C+2;". LINHA?"
   7 INPUT L
   8 IF L>8 THEN PRINT "NO MAXIM
O 8 LETRAS."
   9 IF L>8 THEN GOTO 7
  10 PRINT "A ";C+2;". LINHA TER
A ";L;" LETRAS."
  12 PRINT "ENTRE COM A ";C+2;"
LINHA."
  20 INPUT A$
  24 IF LEN A$<>L THEN PRINT "AT
ENCAO, VOCE ESCOLHEU ";L;" LETRA
S."
  26 IF LEN A$<>L THEN GOTO 20
  27 CLS
  28 PRINT AT 21,7;"AGUARDE UM M
OMENTO"
  29 PAUSE 140
  30 PRINT AT 21,7;"
  "
  31 FAST
  33 IF C=0 THEN RAND USR 16527
  35 LET E=C+1
  40 FOR F=0 TO 7
  50 LET I=960+CODE A$
  60 FOR G=0 TO 7
  70 LET X=PEEK (8*I+G)
  80 FOR H=0 TO 7
  90 LET Z=INT (X/2**(7-H))/2
 100 IF Z<>INT Z THEN PLOT H+8*F
+(8-L)*4,40-G-8*E-2*(C+2)
 110 NEXT H
 120 NEXT G
 130 LET A$=A$(2 TO )
 140 IF LEN A$=0 THEN GOTO 250
 150 NEXT F
 160 NEXT E
 240 SLOW
 250 GOSUB 9100
 285 NEXT C
 286 SLOW
 287 PRINT AT 21,0;"DIGITE <NEW
LINE> P/ LIMPAR TELA"
 288 IF INKEY$<>"" THEN GOTO 291
 289 GOTO 288
 291 CLS
 292 PRINT AT 19,0;"A APRESENTAC
AO FICOU BOA ASSIM?"
 293 PRINT "ENTRE ""S"" SE GOSTO
U DELA"
 294 PRINT "ENTRE ""N"" SE QUIZE
R REFAZE-LA"
 295 IF INKEY$="S" THEN GOTO 310
 300 IF INKEY$="N" THEN GOTO 2
 305 GOTO 295
 310 CLS
 312 LET M=1
 315 GOTO 2000
 500 LET D=7
 502 GOTO 3000
 565 LET X=13
 570 FOR Y=8 TO 15
 575 GOSUB 9980
 580 FOR X=14 TO 16
 585 LET Y=15
 590 GOSUB 9990
 595 LET X=16
 600 FOR Y=14 TO 11 STEP -1
 605 GOSUB 9980
 610 FOR X=15 TO 14 STEP -1
 615 LET Y=11
 620 GOSUB 9990
 625 LET X=19
 630 FOR Y=11 TO 15
 635 GOSUB 9980
 640 PLOT 20,14
 645 PLOT 21,15
 650 LET D=25
 655 GOTO 3200
 725 FOR X=33 TO 30 STEP -1
 730 LET Y=15
 735 GOSUB 9990
 740 LET X=30
 745 FOR Y=14 TO 13 STEP -1
 750 GOSUB 9980
 755 FOR X=31 TO 33
 760 LET Y=13
 765 GOSUB 9990
 770 LET X=33
 775 FOR Y=12 TO 11 STEP -1
 780 GOSUB 9980
 785 FOR X=32 TO 30 STEP -1
 790 LET Y=11
 795 GOSUB 9990
 800 LET D=37
 805 GOTO 3200
 875 LET X=42
 880 FOR Y=15 TO 11 STEP -1
 885 GOSUB 9980
 890 PLOT 43,14
 900 FOR X=44 TO 45
 915 LET Y=15
 920 GOSUB 9990
 925 LET X=46
 930 FOR Y=15 TO 11 STEP -1
 935 GOSUB 9980
 940 LET X=50
 945 FOR Y=17 TO 11 STEP -1
 950 GOSUB 9980
 955 PLOT 49,15
 960 PLOT 51,15
 970 PLOT 51,11
 975 LET D=55
 980 GOTO 3000
2000 CLS
2001 PRINT "O APRESENTADOR",,,,
,
2030 PRINT "VOCE QUER,",,,
2035 PRINT "1. CRIAR UMA APRESEN
TACAO",,,
2040 PRINT "2. FAZER/REFAZER OS
TITULOS",,,
2045 PRINT "3. VER UMA APRESENTA
CAO COMPLETA",,
2050 PRINT "4. VER SO OS TITULOS
"
2055 PRINT "5.""SAVE"" A APRESEN
TACAO SEM/COM  OS TITULOS"
2060 IF INKEY$="1" THEN RUN
2070 IF INKEY$="2" THEN GOTO 800
0
2080 IF INKEY$="3" THEN GOTO 500
0
2083 IF INKEY$="4" THEN GOTO 850
0
2085 IF INKEY$="5" THEN GOTO 700
0
2090 GOTO 2050
2500 GOTO 725
3000 FOR X=D TO D+2
3010 LET Y=15
3020 GOSUB 9990
3030 LET X=D+2
3040 FOR Y=14 TO 11 STEP -1
3050 GOSUB 9980
3060 PLOT D+3,11
3070 FOR X=D+1 TO D-1 STEP -1
3080 LET Y=13
3090 GOSUB 9990
3100 LET X=D-1
3110 FOR Y=12 TO 11 STEP -1
3120 GOSUB 9980
3130 FOR X=D TO D+1
3140 LET Y=11
```

FIGURA 3 (continuação)

```
3150 GOSUB 9990
3160 GOTO D*80
3200 FOR X=D TO D+2
3210 LET Y=13
3220 GOSUB 9990
3230 LET X=D+2
3240 FOR Y=14 TO 15
3250 GOSUB 9980
3260 FOR X=D+1 TO D-1 STEP -1
3270 LET Y=15
3280 GOSUB 9990
3290 LET X=D-1
3300 FOR Y=14 TO 11 STEP -1
3310 GOSUB 9980
3320 FOR X=D TO D+2
3330 LET Y=11
3340 GOSUB 9990
3350 GOTO 100*D
3700 GOTO 875
4400 GOSUB 6000
4405 IF M=1 THEN GOTO 8560
4410 GOTO 8500
5000 SLOW
5002 CLS
5005 FOR C=-1 TO 0
5010 GOSUB 6000
5020 GOSUB 8600
5030 NEXT C
5040 GOSUB 6000
5080 GOTO 500
5090 GOSUB 6000
6000 FOR N=0 TO 10
6010 NEXT N
6020 RETURN
7000 CLS
7001 PRINT " COLOQUE SEU GRAVADO
R NA POSICAO ""REC"", E ENTRE ""
NEWLINE"""
7002 INPUT O$
7003 CLS
7004 LET M=0
7010 LET L$=".."
7020 LET L$(1)=CHR$ PEEK 16404
7030 LET L$(2)=CHR$ PEEK 16405
7040 POKE 16405,113
7050 POKE 16404,0
7060 SAVE "O APRESENTADOR"
7070 POKE 16404,CODE L$(1)
7080 POKE 16405,CODE L$(2)
7085 IF M=1 THEN GOTO 8000
7090 GOTO 2000
8000 CLS
8002 PRINT "TITULOS E SUB-TITULO
S",,
8004 PRINT "QUANTAS TELAS?"
8008 INPUT T
8011 IF T>5 THEN PRINT "ATENCAO,
 NO MAXIMO 5 TELAS."
8012 IF T<1 OR T>5 THEN GOTO 800
8
8015 PRINT T
8020 FOR C=1 TO T
8030 PRINT "QUANTAS LINHAS TERA
A ";C;". TELA?"
8038 INPUT W
8040 IF W>4 THEN PRINT "ATENCAO,
 NO MAXIMO 4 LINHAS."
8042 IF W>4 THEN GOTO 8038
8043 PRINT W
8050 FOR E=1 TO W
8060 PRINT "QUANTAS LETRAS TERA
A ";E;". LINHA?"
8070 INPUT L
8072 IF L>8 THEN PRINT "ATENCAO,
 NO MAXIMO 8 LETRAS."
8073 IF L>8 THEN GOTO 8070
8075 PRINT L
8160 PRINT "ENTRE COM A ";E;". L
INHA                  "
8170 INPUT A$
8175 IF LEN A$<>L THEN PRINT "AT
ENCAO, VOCE ESCOLHEU ";L;" LETRA
S."
8177 IF LEN A$<>L THEN GOTO 8170
8178 CLS
8180 FAST
8190 IF E>1 THEN GOSUB 8600
8200 FOR F=0 TO L
8210 LET I=960+CODE A$
8220 FOR G=0 TO 7
8230 LET X=PEEK (8*I+G)
8240 FOR H=0 TO 7
8250 LET Z=INT (X/2**(7-H))/2
8260 IF Z<>INT Z THEN PLOT H+8*F
+(8-L)*4,49-INT (48/(W+1))-INT (
64/(W+2))*(E-1)-G
8270 NEXT H
8280 NEXT G
8290 LET A$=A$(2 TO )
8300 IF LEN A$=0 THEN GOTO 8311
8310 NEXT F
8311 SLOW
8315 GOSUB 9100
8316 IF E<>W THEN CLS
8317 NEXT E
8318 PRINT AT 21,0;"DIGITE <NEW
LINE> P/ LIMPAR TELA"
8319 INPUT X$
8320 CLS
8322 PRINT AT 19,0;"A ";C;". TEL
A ESTA BOA ASSIM?"
8323 PRINT """S"" NO CASO AFIRMA
TIVO"
8324 PRINT """N"" SE QUIZER REFA
ZER A TELA"
8325 IF INKEY$="S" THEN GOTO 833
0
8327 IF INKEY$="N" THEN GOTO 840
0
8329 GOTO 8325
8330 CLS
8331 NEXT C
8335 SLOW
8336 LET M=2
8340 PRINT "VEJA COMO FICOU,",,,
"DEPOIS ENTRE ""M"", PARA RETORN
AR AO MENU"
8350 GOSUB 6000
8355 GOSUB 6000
8360 GOTO 5000
8400 CLS
8410 GOTO 8030
8500 CLS
8501 IF M=1 THEN GOTO 2000
8505 FOR C=1 TO T
8510 GOSUB 8600
8540 GOSUB 6000
8550 NEXT C
8560 IF INKEY$="M" THEN GOTO 200
0
8561 GOTO 8560
8562 CLS
8565 GOTO 2000
8570 STOP
8580 GOTO 5000
8600 POKE 16507,128
8610 POKE 16508,94+3*C
8620 RAND USR 16527
8630 RETURN
8990 STOP
9100 POKE 16507,128
9110 POKE 16508,94+3*C
9120 RAND USR 16514
9140 RETURN
9980 PLOT X,Y
9981 NEXT Y
9982 RETURN
9990 PLOT X,Y
9991 NEXT X
9992 RETURN
```

Se você escolher a opção nº 4 verá títulos sem a apresentação inicial.

Interrompa a gravação quando a apresentação estiver gravada (se o seu vídeo cassete tiver o recurso "AUDIO DUB" você poderá gravar só a imagem da apresentação, e gravar o som posteriormente com calma).

Após ter gravado os títulos, retorne a fita do vídeo ao início e veja se a obra ficou boa. Você sempre poderá repetir tudo se não tiver gostado.

Uma vez satisfeito, tire a fita e carregue-a (não preciso dizer que você deve começar a gravar a fita após o ponto onde o título foi gravado, para não gravar sobre ele e perdê-lo).

Vamos à festa,que a família já está esperando. Leve seu filho como iluminador e mãos à obra.

Não coma muitos doces que você está a serviço e além disso fazendo regime. TRABALHE.

Após a festa, volte à casa do Marcelo e mostre sua obra a todos. Você será nomeado o "Dias Gomes" da família.

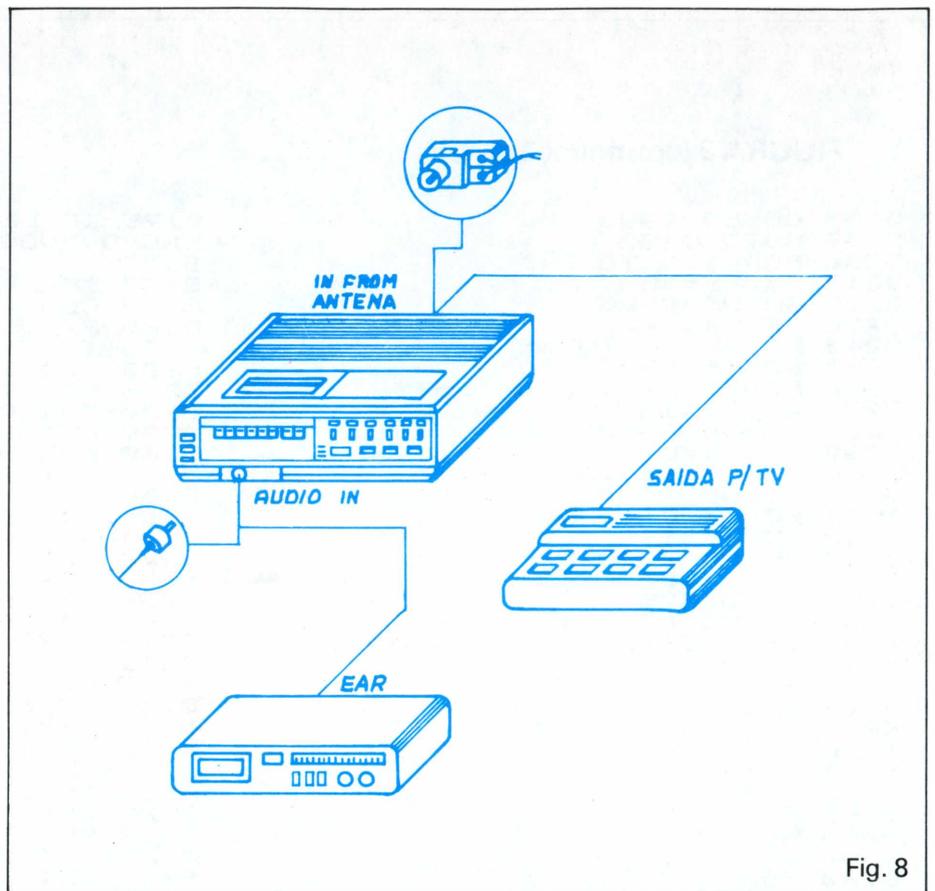



Fig. 8

Torneio de Xadrez

Renato da Silva Oliveira

Numa das últimas cartas que Nabor me enviou, encontrei novamente um interessante problema. Sua solução não é das mais difíceis, entretanto, o que nos interessa é um programa para resolvê-lo, e isso é um pouco mais complicado.

O trecho da carta que nos interessa está reproduzido adiante.

"Como você sabe, sempre me interessei muito por xadrez. Todas as noites, eu, Tolen, Ramarujan e Harold temos jogado até altas horas da madrugada.

A alguns dias atrás, ensinei os movimentos das peças e algumas estratégias de jogo para a Dinorá. De início, ela teve muita dificuldade em assimilar o jogo mas, graças aos morangos silvestres de Ramarujan, em pouco tempo estava jogando tão bem quanto eu. Os efeitos desses morangos sobre a Dinorá estão cada vez mais notórios, porém, de vez em quando ela tem uma recaída e eu nem a reconheço, ou melhor, tenho medo de conhecê-la.

Numa dessas vezes, ela teve a idéia de realizar um torneio de xadrez entre nós cinco, e inventou um critério esdrúxulo para eliminação dos participantes. Na verdade, realizamos dois desses torneios. Nenhum de nós se opôs ao critério para não correr o risco de magoar a Dinorá.

Cada torneio teve quatro partidas. Os jogadores da primeira partida foram escolhidos por sorteio. O perdedor foi eliminado e o vencedor jogou a segunda partida com um terceiro, sorteado entre os três que ainda não haviam jogado. Novamente, o perdedor da segunda partida foi eliminado e o vencedor jogou a terceira partida com um quarto participante, sorteado entre os dois restantes. Finalmente, a quarta e última partida foi realizada entre o vencedor da terceira e o quinto e último participante. O vencedor dessa quarta partida foi o vencedor do torneio.

No 1º torneio que disputamos, as partidas foram as seguintes:

Eu contra Tolen, Eu contra Harold, Ramarujan contra Dinorá, Ramarujan contra Harold.

Obviamente, não as realizamos na ordem em que eu as apresentei.

Com um pouco de sorte, consegui vencer o primeiro torneio.

Uma surpreendente coincidência fez com que as partidas do segundo torneio fossem realizadas com as mesmas duplas do primeiro torneio, com a diferença de que os participantes que haviam vencido apenas uma partida no primeiro torneio, perderam para o mesmo adversário no segundo.

Com isso, como é evidente, o venc. . .

Aqui está seu quebra-cabeça: quem venceu o 2º torneio? Responda a isso com um programa.

P.S. Continuamos a aguardar a resposta do Quebra-Cabeça "O amor do Sr. Nabor", que saiu na edição número 11 da Microhobby!

Apresentamos o TK 2000 II. Ele roda o programa mais famoso do mundo.

De hoje em diante nenhuma empresa, por menor que seja, pode dispensar o TK 2000 II. Por que?

O novo TK 2000 II roda o Multicalc: a versão Microsoft do Visicalc®, o programa mais famoso em todo o mundo.

Isto significa que, com ele, você controla estoques, custos, contas a pagar, faz sua programação financeira, efetua a folha de pagamentos e administra minuto a minuto as suas atividades.

Detalhe importante: o novo TK 2000 II, com Multicalc, pode intercambiar planilhas com computadores da linha Apple®.

E, como todo business computer que se preza, ele tem teclado profissional, aceita monitor, diskette, impressora e já vem com interface.

Além de poder ser ligado ao seu televisor (cores ou P&B), oferecendo som e imagem da melhor qualidade.

Portanto, peça logo uma demonstração do novo TK 2000 II, nas versões 64K ou 128K de memória.

A mais nova estrela do show business só espera por isto para estrear no seu negócio.

Preço de lançamento* (128 K):
Cr$ 1.949.850

MICRODIGITAL
computadores pessoais

Open for Business.

® Marca registrada da Apple Computer.

Filiada à ABICOMP

® Marca registrada da Visicorp.

* Sujeito a alteração sem prévio aviso.

CALCULADORAS

ANÁLISE DE FOURIER
PROGRAMA APLICATIVO

Jose Eduardo Moreira
Wilson José Tucci

Se uma função f(x) é "bem comportada", podemos escrevê-la como uma soma infinita de senos e cossenos:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos \frac{n\pi}{L} x + b_n \operatorname{sen} \frac{n\pi}{L} x)$$

onde: $a = \frac{1}{L} \int_{-L}^{L} f(x) \cos \frac{n\pi}{L} x\, dx$

$$b = \frac{1}{L} \int_{-L}^{L} f(x) \operatorname{sen} \frac{n\pi}{L} x\, dx$$

Essa igualdade é válida para o intervalo] − L, L [, ou se f(x) for periódica de período 2L, vale para o intervalo] −∞, +∞[.

Obviamente não podemos calcular numericamente uma soma dos infinitos termos; o que se faz, então, é aproximar f(x) pelos primeiros termos da soma infinita, assim:

$$f(x) \frac{a^o}{2} + \sum_{n=1}^{m} (a_n \cos \frac{n\pi}{L} x + b_n \operatorname{sen} \frac{n\pi}{L} x)$$

O que faremos aqui será desenvolver um programa que, para uma dada f(x), um certo L e um certo m, calcule os coeficientes a_0, a_1, . . . a_m e b_1, b_2, . . ., b_m que aproximam f(x) no intervalo] − L, L [.

O algoritmo para o cálculo desses coeficientes pode ser escrito assim:

1. leia L, m, d
2. $a_0 \leftarrow \frac{1}{L} \int_{-L}^{L} f(x)\, d(x)$
3. mostre a_0
4. repita os passos 4.1 a 4.4 para n de 1 até m

   4.1 $a_n \leftarrow \frac{1}{L} \int_{-L}^{L} f(x) \cos \frac{n\pi}{L} x\, dx$

   4.2 mostre a_n

   4.3 $b_n \leftarrow \frac{1}{L} \int_{-L}^{L} f(x) \operatorname{sen} \frac{n\pi}{L} x\, dx$

   4.4 mostre b_n

Para calcular as integrais, utilizaremos uma versão modificada do algoritmo para a regra de Simpson (veja o último número).

para os a_n:

1. h ← 2L/d
2. SOMA ← f(− L) * cos n π + 4 * f(− L h) * cos $\frac{n\pi}{L}$ (− L + h) + f(L) * cos n π
3. repita o passo 3.1 para j de 2 até d−2 de 2 em 2

   3.1 SOMA ← SOMA + 2 * f(− L + jh) * cos $\frac{n\pi}{L}$ (−L + jh) + 4 * f(− L + (j + 1) h) * cos $\frac{n\pi}{L}$ (− L + (j + 1) h)

4. SIMP ← h/3 * SOMA
5. Mostre SIMP

Para os b_n:

1. h ← 2L/d
2. SOMA ← 4 * f(− L + h) * sen $\frac{n\pi}{L}$ (− L + h)
3. repita o passo 3.1 para j de 2 até d−2 de 2 em 2

   3.1. SOMA ← SOMA + 2 * f(− L + jh) * sen $\frac{n\pi}{L}$ (− L + jh) + 4 * f(− L + (j + 1) h) * sen $\frac{n\pi}{L}$ (− L + (j + 1) h)

4. SIMP ← h/3 * SOMA
5. SIMP

Nas figuras 1 e 2, vemos dois exemplos com funções diferentes. Nas tabelas I e II podemos ver os comentários do programa.

L = 1, M = 7, d = 20 (tem que ser par)

A0 = 1.0000
A1 = − 0.4052
B1 = -3.3333×10^{-11}
A2 = 3×10^{-10}
B2 = 0.0000
A3 = − 0.0443
B3 = -3.3333×10^{-11}

Figura 1

L = 1, M = 7, d = 20 (tem que ser par)

A0 = 0.0000
A1 = 6.6667×10^{-11}
B1 = 0.6367
A2 = 3.3333×10^{-11}
B2 = − 03186
A3 = -1.0000×10^{-10}
B3 = 0.2132
A4 = 0.0000
B4 = − 0.1619
A5 = -1.1947×10^{-9}
B5 = 0.1333
A6 = 0.0000
B6 = − 0.1185
A7 = -3.3333×10^{-11}
B7 = 0.1164

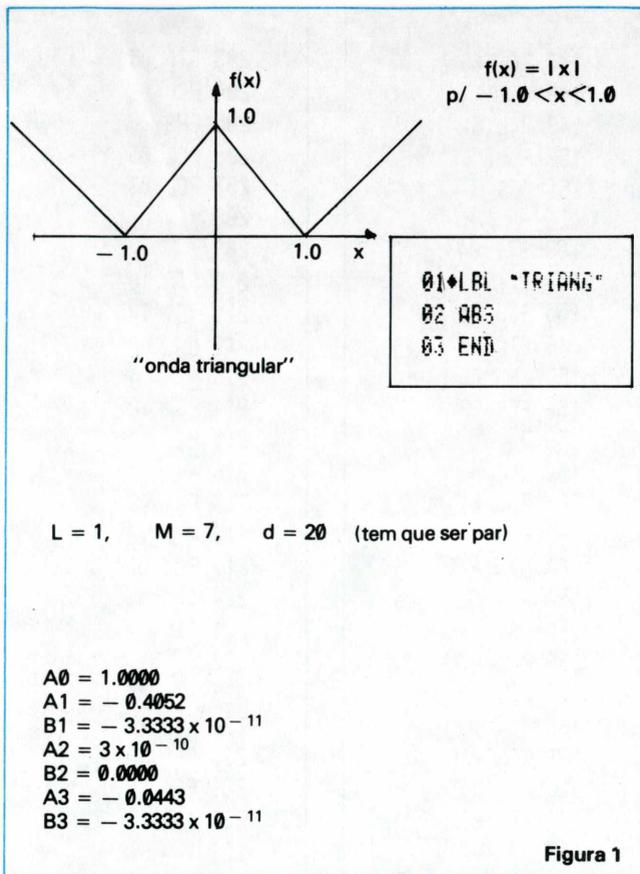
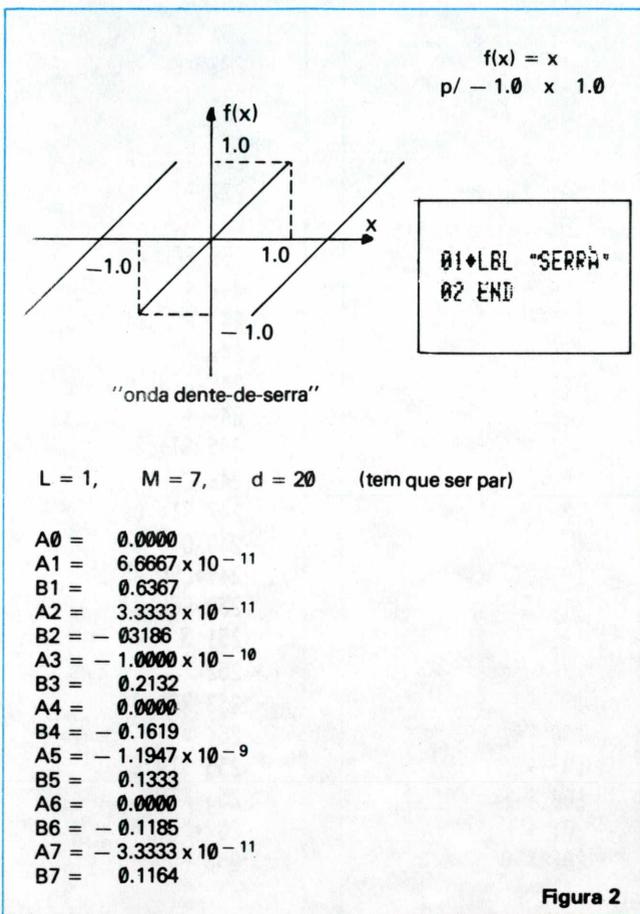



Figura 2

Tabela I

Comentários do programa:

registrador	variável	função
00	m	limite dos índices dos coeficientes
01	L	meio período para a análise
02	d	intervalos para integração
03	h	amplitude dos intervalos
04	SOMA	resultado parcial da integral
05	j	contador para integral
06	f(x)	nome da função
07	SIMP	resultado da integral
08	n	contador dos coeficientes

Tabela II

linhas 02-07:	entrada do nome da função a ser analisada
linhas 08-10:	entrada de L (metade do período em que a função será analisada — L, L)
linhas 11-13:	entrada de M (índice do último coeficiente a ser obtido)
linhas 14-16:	entrada de d (número de intervalos a ser usado na integração numérica), tem que ser par
linhas 17-24:	cálculo e apresentação de a_0
linhas 25-26:	inicaliza n em 1, para o laço das linhas 27-58
linhas 27-58:	laço para o cálculo dos a_n, n = 1, 2, . . . m
linhas 28-32:	comparação para ver se continua no laço ou escapa
linhas 33-42:	cálculo e apresentação de a_n
linhas 43-52:	cálculo e apresentação de b_n
linhas 53-54:	incrementa n
linha 55:	volta
linhas 56-164:	sub-rotina para calcular $\int_{-L}^{L} f(x) \cos \frac{n \pi}{L} x \, dx$
linhas 165-256:	sub-rotina para calcular $\int_{-L}^{L} f(x) \text{ sen } \frac{n \pi}{L} x \, dx$
linhas 257-258:	final

```
01♦LBL "FOURIER"
02 CLA
03 "FUNCAO ?
04 AON
05 PROMPT
06 ASTO 06
07 AOFF
08 "L ?"
09 PROMPT
10 STO 01
11 "M ?"
12 PROMPT
13 STO 00
14 "d ?"
15 PROMPT
16 STO 02
17 0
```

```
18 STO 08
19 XEQ "COSSIMP"
20 RCL 01
21 /
22 "A0="
23 ARCL X
24 PROMPT
25 1
26 STO 08
27♦LBL 00
28 RCL 08
29 RCL 00
30
31 X>0?
32 GTO 01
33 XEQ "COSSIMP
34 RCL 01
```

```
35 /
36 "A"
37 FIX 0
38 ARCL 08
39 FIX 4
40 "F="
41 ARCL X
42 PROMPT
43 XEQ "SENSIMP"
44 RCL 01
45 /
46 "B"
47 FIX 0
48 ARCL 08
49 FIX 4
50 "F="
51 ARCL X
52 PROMPT
53 1
54 ST+ 08
55 GTO 00
56♦LBL "COSSIMP"
57 0
58 STO 04
59 RCL 01
60 2

62 RCL 02
63 /
64 STO 03
65 RCL 01
66 CHS
67 XEQ IND 06
68 RCL 08
69 PI
70 *
71 COS
72 *
73 ST+ 04
74 RCL 01
75 CHS
76 RCL 03
77 +
78 XEQ IND 06
79 RCL 08
80 PI
81 *
82 RCL 01
83
84 RCL 01
85 CHS
86 RCL 03
87 +
88 *
89 COS
90 *
91 4
92 *
93 ST+ 04
94 RCL 01
95 XEQ IND 06
96 RCL 08
97 PI
98 *
99 COS
100 *
101 ST+ 04
102 2
103 STO 05
104♦LBL 02
105 RCL 05
106 RCL 02
107 -
108 2
109 +
110 X>0?
111 GTO 03
112 RCL 01
113 CHS
114 RCL 05
115 RCL 03
116 *
117 +
118 STO 10
119 XEQ IND 06
120 2
121 *
122 RCL 10
123 PI
124 *
125 RCL 08
126 *
127 RCL 01
128 /
129 COS
130 *
131 ST+ 04
132 RCL 01
133 CHS
134 RCL 05
135 1
136 +
137 RCL 03
138 *
139 +
140 STO 10
141 XEQ IND 06
142 4
143 *
144 RCL 10
145 PI
146 *
147 RCL 08
148 *
149 RCL 01
150 /
151 COS
152 *
153 ST+ 04
154 2
155 ST+ 05
156 GTO 02
157♦LBL 03
158 RCL 03
159 3
160 /
161 RCL 04
162 *
163 STO 07
164 RTN
165♦LBL "SENSIMP"
166 RCL 01
167 2
168 *
169 RCL 02
170 /
171 STO 03
172 0
173 STO 04
174 RCL 01
175 CHS
176 RCL 03
177 +
178 XEQ IND 06
179 4
180 *
181 RCL 01
182 CHS
183 RCL 03
184 +
185 PI
186 *
187 RCL 08
188 *
189 RCL 01
190 /
191 SIN
192 *
193 ST+ 04
194 2
195 STO 05
196♦LBL 04
197 RCL 05
198 RCL 02
199 -
200 2
201 +
202 X>0?
203 GTO 05
204 RCL 01
205 CHS
206 RCL 05
207 RCL 03
208 *
209 +
210 STO 10
211 XEQ IND 06
212 2
213 *
214 RCL 10
215 PI
216 *
217 RCL 08
218 *
219 RCL 01
220 /
221 SIN
222 *
223 ST+ 04
224 RCL 01
225 CHS
226 RCL 05
227 1
228 +
229 RCL 03
230 *
231 +
232 STO 10
233 XEQ IND 06
234 4
235 *
236 RCL 10
237 PI
238 *
239 RCL 08
240 *
241 RCL 01
242 /
243 SIN
244 *
245 ST+ 04
246 2
247 ST+ 05
248 GTO 04
249♦LBL 05
250 RCL 03
251 3
252 /
253 RCL 04
254 *
255 STO 07
256 RTN
257♦LBL 01
258 END
```

Curso de Programação (aula IV)

Jose Eduardo Moreira
Wilson José Tucci

Na aula passada foram apresentados alguns conceitos de programação e fluxogramação. Já discutimos a validade e a importância dos fluxogramas e é importante que você se acostume com eles. Mas a nossa HP-41 (e computadores em geral) não aceitam programas na forma de fluxograma. O que fazer então?

A etapa seguinte à fluxogramação é a codificação, um processo bastante mecânico, em que o fluxograma é traduzido para uma determinada linguagem (o fluxograma deve ser o mais genérico possível, válido para qualquer computador, já que as linguagens são específicas de um ou outro computador).

Para podermos traduzir os fluxogramas para a linguagem da HP-41, precisamos aprender um pouco sobre a pilha operacional, os registradores, os comandos de interrupção, os comandos de comparação e o comando GTO.

A Pilha Operacional

A HP-41 possui 4 registradores especiais que formam a pilha operacional. São eles os registradores X, Y, Z e T (figura 1).

Figura 1

T	
Z	
Y	
X	único apresentado no visor

A pilha operacional é parte fundamental da implementação da RPN nas calculadoras HP.

Quando se digita um número no teclado, (por exemplo: 5), ele é colocado no registrador X sem que haja mudança alguma nos conteúdos dos registradores Y, Z e T (figura 2).

Figura 2

T	0.0000		T	0.0000
Z	0.0000	ENTER↑	Z	0.0000
Y	0.0000		Y	5.0000
X	5		Y	5.0000

Ao acionarmos a tecla ENTER↑, a pilha inteira sobe (o que estava em T é perdido, o que estava em Z passa a T, o que estava em Y passa a Z e o que estava em X passa para Y) e o que estava em X, continua lá. Quando alguma operação é executada, ela é sempre do tipo Y operação X. O resultado é colocado em X e a pilha desce (o que estava em Z passa para Y, o que estava em T passa para Z e continua em T). Acompanhe na figura 3 o estado da pilha durante o cálculo da expressão (783 * 435)/329.

Tecla		Registrador	Conteúdo
XEQ	CLST	T	0.0000
		Z	0.0000
		Y	0.0000
		X	0.0000
783		T	0.0000
		Z	0.0000
		Y	0.0000
		X	783
ENTER ↑		T	0.0000
		Z	0.0000
		Y	783.0000
		X	783.0000
435		T	0.0000
		Z	0.0000
		Y	783.0000
		X	435
*		T	0.0000
		Z	0.0000
		Y	0.0000
		X	340605.0000
329		T	0.0000
		Z	0.0000
		Y	340605.0000
		X	329
/		T	0.0000
		Z	0.0000
		Y	0.0000
		X	1035.2736

Figura 3

Registradores

A HP-41 apresenta um grande número de registradores, que pode variar desde 63 (HP-41C "seca"), até um máximo de 319 registradores (HP-41C + 4 módulos, HP-41C + Quad Memory, HP-41 CV, HP-41CX). Todos esses registradores podem ser usados para guardar dados (números, letras), ou programas. A repartição dos registradores é feita pelo usuário através do comando SIZE (XEQ S I Z E), e especificando o número de registradores que devem ser usados para dados (os restantes são reservados para programas). Cada variável do fluxograma terá um registrador de dados equivalente na codificação. Após dar um SIZE, estão disponíveis os registradores de número 00 até o número menos uma unidade, que é o número especificado no SIZE.

Os registradores são operados fundamentalmente através das operações:
STO — Store (armazena)
RCL — Recall (chame de novo)

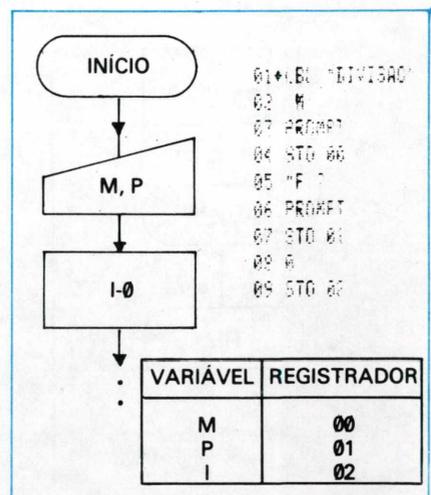
A operação STO "armazena no registrador de número" o número (ou letra) guardado no registrador X.

A operação RCL chama para o registrador X, o valor (número ou letras) guardado no registrador".

Os comandos de interrupção servem para interromper a execução do programa, para que se possa entrar com dados ou ler as saídas. São dois os principais:
STOP — (tecla R/S), interrompe a execução do programa e visualiza o conteúdo do registrador X. Para continuar a execução do programa, aperte R/S.
PROMPT — (XEQ PROMPT), interrompe a execução do programa e visualiza o conteúdo do registrador ALPHA. Para continuar a execução do programa, tecle R/S.

Com o que já sabemos, podemos traduzir o começo do fluxograma da aula passada (figura 4).

Figura 4

Fluxograma

Comandos de comparação

A HP-41 dispõe de cinco comandos para comparação entre os valores dos registradores X e Y e mais cinco comandos para comparação entre o registrador X e o valor 0. As comparações possíveis estão mostradas na Tabela I.

Tabela I

X com Y	X com 0
X = Y?	X = 0?
X = Y?	X = 0
X < Y?	X < 0?
X < = Y?	X < = 0?
X > Y?	X > 0?

Se a comparação for verdadeira, o comando seguinte é executado; se for falsa, o comando seguinte é pulado.

Assim, ao traduzirmos uma operação de decisão, faremos a comparação que, se verdadeira, força o computador a desviar; a seguir colocamos uma instrução de desvio (GTO) e depois a continuação normal do fluxograma.

O controle GTO LL força um desvio para o LBL LL. Agora podemos acabar de traduzir nosso fluxograma (figura 5).

Exercícios

Traduza os fluxogramas dos exercícios da aula passada.

Fluxograma

Tradução/Codificação

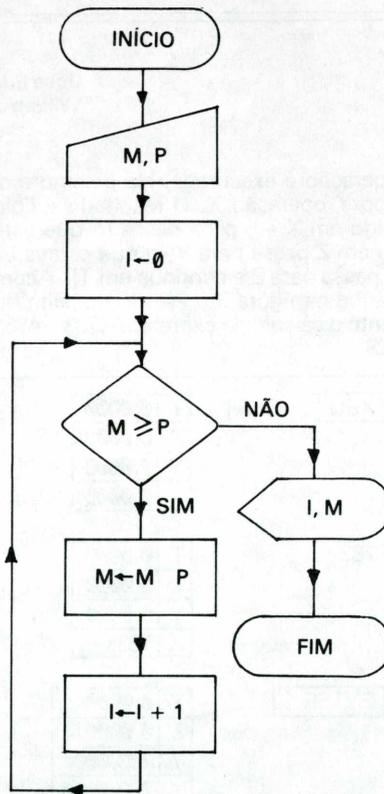



Figura 5

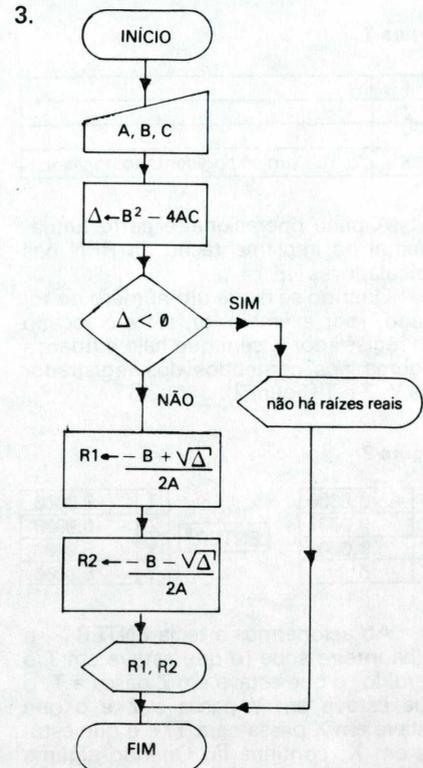
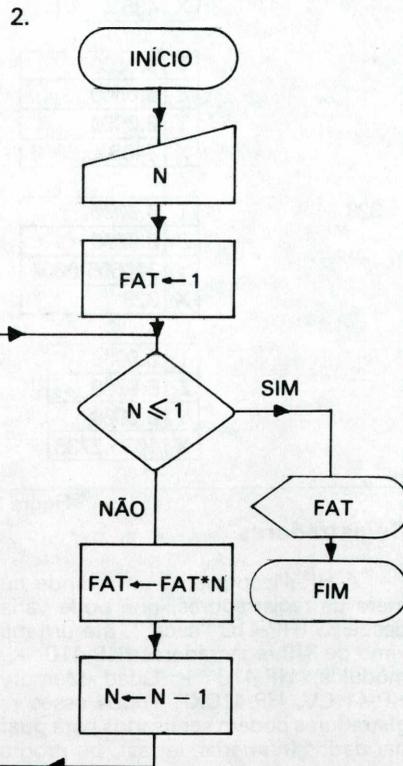
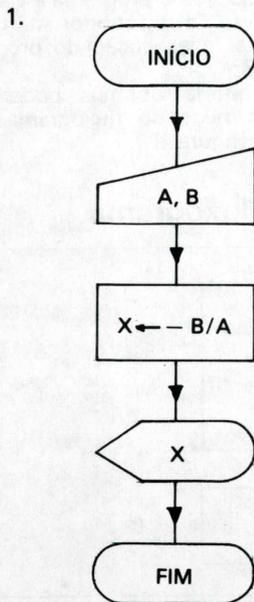
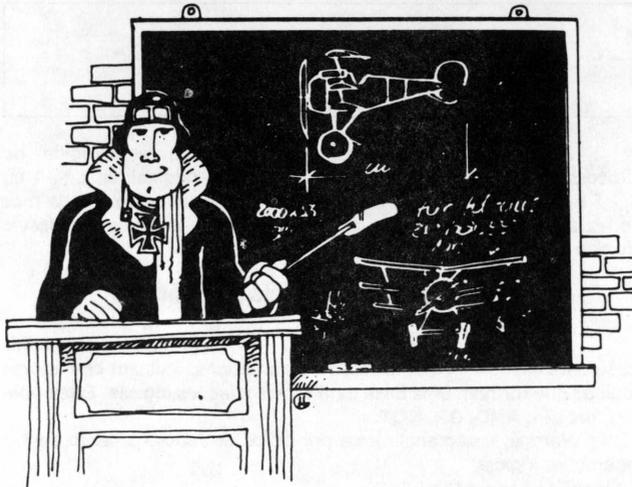
```
01♦LBL "DIVISAO
02 "M ?"
03 PROMPT
04 STO 00
05 "P ?"
06 PROMPT
07 STO 01
08 0
09 STO 02
10♦LBL 00
11 RCL 00
12 RCL 01
13 X>Y?
14 GTO 01
15 RCL 00
16 RCL 01
17 -
18 STO 00
19 RCL 02
20 1
21 +
22 STO 02
23 GTO 00
24♦LBL 01
25 "QUOCIENTE ="
26 ARCL 02
27 PROMPT
28 "RESTO ="
29 ARCL 00
30 PROMPT
31 END
```

VARIÁVEL	REGISTRADOR
M	00
P	01
L	02

Respostas dos exercícios da aula anterior:

1.

2.

3.

Traduções para os TK 83/85: Uma nova visão

Mario Cezar Barbosa

Uma das grandes desvantagens dos microcomputadores da linha TK é exatamente a inexistência dos comandos de desvios condicionados, do tipo ON k GOTO x, y, z, . . . e ON k GOSUB x, y, z. . . ., onde k é uma variável condicional inteira e x, y, z são números de linhas para desvios. Assim, se k = 1 o desvio faz-se para a linha x; se k = 2 o desvio faz-se para a linha y; e sucessivamente.

Existem, entretanto, diversos "macetes" que simulam estes comandos de desvio condicionado. Dentre esses "macetes", destaco os apresentados no excelente artigo de Igor Sartoni (MICROHOBBY Nº 3-1983). Entretanto tais artifícios têm uma grande desvantagem: ou não são aplicáveis em todos os casos ou ocupam muita memória.

Como, então, implementar no TK os comandos de desvio condicionados, utilizando-se pouca memória e que sejam mais abrangentes?

A solução que proponho é a utilização das operações aritméticas lógicas, que seu TK pode fazer.

O que é uma operação Lógica?

Uma operação lógica é aquela ocorrida "entre duas coisas" e que apresenta como resultado possível somente dois estados: VERDADEIRO (1) ou FALSO (Ø).

Assim, por exemplo, se consideramos o operador lógico >, e queremos operar 5 com 3, nesta ordem, com esse operador, podemos escrever:

operação	resposta	
5 > 3	1	(verdadeiro)

Ao contrário, se utilizamos o operador lógico =, com estes mesmos números, obtemos:

operação	resposta
5 = 3	Ø

O seu TK tem capacidade de realizar operações deste tipo com os seguintes "OPERADORES LÓGICOS": <, >, = , < =, > =, < >,

Por exemplo, ligue o seu TK e pergunte a ele:

```
PRINT 2=7
PRINT 5=5
PRINT 9>1
```

observe as respostas oferecidas pelo computador: Ø (falso) e 1 (verdadeiro).

Equações Lógicas

Assim como ocorre com os operadores +, –, /, *, podemos realizar "operações" com os operadores lógicos e os operadores numéricos. Desta forma ampliamos nosso quadro de operações matemáticas para: +, –, /, *, <, >, =, < =, > =, < >.

Em seu TK são permitidas operações mistas com "operadores lógicos" e "numéricos ou aritméticos". Assim, podemos escrever expressões do tipo:

```
10 LET Z=5+3*(2=1)
20 LET A=4*(Z<8)
30 PRINT "Z=";Z,"A=";A
```

observe os resultados da operação: Z = 5 e A = 4

Analisando-se as equações observamos que: para Z, inicialmente, o TK calculou a operação lógica 2 = 1 e encontrou Ø; depois operou Ø*3 + 5 e obteve 5 como resposta; para A, inicialmente, o TK operou (Z < 8) e encontrou 1 (verdadeiro) e depois operou 4*1 encontrando 4 como resposta. Interessante, não é?

Agora você já pode elaborar equações lógico-aritméticas em seu TK, experimente!

Tente, agora, realizar equações lógicas com caracteres alfanuméricos por exemplo:

```
10 PRINT "A"="B"
20 PRINT "A"<"B"
30 PRINT "C">"D"
```

observe as respostas e conclua que a comparação de alfanuméricos está baseada na ordenação alfabética.

Vamos, agora, estudar alguns "macetes" de programação baseados nas "equações lógico-aritméticas".

Substituições de comandos do tipo: IF X < Y THEN LET Z = Z + N

onde: X, Y, Z, são variáveis ou números e N é uma variável tipo numérica; 01 é um "operador lógico".

Podemos substituir essa expressão por:

LET Z = Z + N* (Z < Y)

por exemplo, seja:

```
100 IF X=Y THEN LET RAMO=RAMO+3
```

fica como:

```
100 LET RAMO=RAMO+3*(X=Y)
```

a equação lógica (X + Y) assume os valores Ø (para falso) e 1 (para verdadeiro), de forma que 3*(X = Y) será 3 (se X = Y) ou Ø (se X<>Y).

As vantagens da substituição são: maior rapidez de cálculos e economia de 1 byte de memória.

Implementação dos comandos de Desvio Condicionado: ON k GOTO x, y, z, e ON k GOSUB x, y, z,.

Podemos utilizar "equações lógicas" para a "simulação" destes comandos no TK.

Para tal, utilizamos a seguinte equação:

Substituimos:

```
ON K GOTO X,Y,Z,...
```

por:

```
GOTO X*(K=1)+Y*(K=2)+Z*(K=3)+...
```

e analogamente substituimos:

```
ON K GOSUB X,Y,Z,...
```

por:

```
GOSUB X*(K=1)+Y*(K=2)+Z*(K=3)+..
```

Claramente, apenas uma das expressões lógicas nos parênteses será verdadeira (1) e o desvio se fará para a linha multiplicativa; as demais equações lógicas assumem o valor zero (Ø) e não interferem no desvio. Observe que não fizemos restrições aos números x, y, z, . . . das linhas de desvio, e o processo faz-se então genericamente.

As vantagens do processo são: economiza memória em muitos casos (principalmente se utilizarmos o poder de VAL nas constantes numéricas), grande rapidez de cálculo e desvio. A principal desvantagem do processo deve-se ao fato de que os comandos de desvio condicionado não são executados quando *k* vem com um valor fora do intervalo utilizado para o desvio; corrigiremos esta deficiência mais adiante, depois de estudarmos as operações de associação lógica do TK.

Como exemplos de desvios condicionados, temos:

```
100 ON K GOTO 20,100,13,55,89
```

substituimos por:

```
100 GOTO 20*(K=1)+100*(K=2)+13*(
K=3)+55*(K=4)+89*(K=5)
```

e, se desejamos economizar memória, escrevemos:

```
100 GOTO VAL "20*(K=1)+100*(K=2)
+13*(K=3)+55*(K=4)+89*(K=5)"
```

Podemos observar aqui a desvantagem, a ser corrigida, do processo: enquanto na linha 1ØØ do programa original, se k < 1 ou k > 5 a linha 1ØØ não seria executada, em nosso processo obtería-mos todas as expressões lógicas como falsas (Ø) e teríamos um desvio do tipo GOTO Ø.

Operações de Associação Lógica

Assim como os operadores +, −, /, * formam as operações base para associações de elementos numéricos, existem operadores lógicos que formam uma base para as associações lógicas. Estes operadores são: AND, OR, NOT.

Vamos, então analizar os princípios de funcionamento destas operações lógicas:

Operação Lógica AND:

A operação Lógica AND traduz-se por "E", e apresenta um significado de REUNIÃO e SIMULTANEIDADE. Assim, se efetuamos a operação Lógica: A AND B, onde A e B são expressões lógicas, temos como resultado o seguinte quadro:

TABELA 1

A	B	A AND B
1	1	1
1	Ø	Ø
Ø	1	Ø
Ø	Ø	Ø

onde: 1 significa operação verdadeira e Ø falsa.

Por exemplo, ligue o seu TK e execute:

```
10 PRINT 1<2 AND 5<6
20 PRINT 1<2 AND 4>5
30 PRINT 9<1 AND 8>4
40 PRINT 1=2 AND 3=2
```

Operação Lógica OR:

A operação lógica OR é traduzida como "OU", e apresenta um significado de INDEPENDÊNCIA. Assim, se efetuamos a operação lógica: A OR B, onde A e B são expressões lógicas, temos como resultado o seguinte quadro:

TABELA 2

A	B	A OR B
1	1	1
1	Ø	1
Ø	1	1
Ø	Ø	1

Por exemplo, ligue o seu TK e execute:

```
10 PRINT 1<2 OR 5<6
20 PRINT 1<2 OR 4>5
30 PRINT 9<1 OR 8>4
40 PRINT 1=2 OR 2=3
```

Operação Lógica NOT:

A operação lógica NOT traduz-se por "NÃO", e apresenta um significado de NEGAÇÃO. Podemos formar as seguintes tabelas com NOT:

VER TABELA

TABELA 3

A	NOT A
1	0
0	1

Por exemplo, ligue o seu TK e execute:

```
10 PRINT NOT (2=2)
20 PRINT NOT (1=2)
```

outra associação possível é:

TABELA 4

A	B	A AND B	NOT (A AND B)
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	1

Ligue o seu TK e execute:

```
  10 PRINT 1<2 AND 5<6,NOT (1<2
AND 5<6)
  20 PRINT 1<2 AND 4>5,NOT (1<2
AND 4>5)
  30 PRINT 9<1 AND 8>4,NOT (9<1
AND 8>4)
  40 PRINT 1=2 AND 2=3,NOT (1=2
AND 2=3)
```

e, finalmente temos a seguinte associação:

TABELA 5

A	B	A OR B	NOT (A OR B)
1	1	1	0
1	0	1	0
0	1	1	0
0	0	0	1

Ligue o seu TK e execute:

```
10 PRINT 1<2 OR 5<6,NOT (1<2
OR 5<6)
20 PRINT 1<2 OR 4>5,NOT (1<2
OR 4>5)
30 PRINT 9<1 OR 8>4,NOT (9<1
OR 8>4)
40 PRINT 1=2 OR 2=3,NOT (1=2
OR 2=3)
```

Outras tabelas lógicas poderiam ainda ser elaboradas com o operador NOT, tais como; NOT(A) AND B; NOT(A) OR B; A AND NOT(B); A OR NOT(B); NOT(A) AND NOT(B); NOT(A) OR NOT(B); NOT (NOT(A) AND B) e outras. Entretanto, estas tabelas podem ser facilmente elaboradas com auxílio das anteriores e do TK, e em geral são raramente utilizadas em programação.

Variáveis Lógicas

Além dos tipos de variáveis usuais (numéricas e alfanuméricas), os TKs possuem ainda as denominadas "Variáveis Lógicas".

Uma variável lógica é uma variável que assume unicamente dois estados: 1 (verdadeiro) e 0 (falso). Esse tipo de variável apresenta uma importância muito grande como variável de controle de processos elaborados em programas mais complexos.

Para observarmos as variáveis lógicas dos TK, consideremos os seguintes programas:

```
10 LET X=400
20 PRINT NOT X
```

como resultado deste programa obteremos o nível lógico 0 (falso), e:

```
10 LET X=0
20 PRINT NOT X
```

como resultado deste, obtemos o nível lógico 1 (verdadeiro).

Na realidade o que fizemos foi o seguinte: definimos uma variável numérica X nos programas e pedimos a impressão de uma variável lógica NOT X, correspondente.

Uma observação atenta deste processo, nos conduz ao seguinte resultado:

```
Se X<>0 então NOT X = 0

Se X = 0 então NOT X = 1
```

Assim se definimos em um programa uma variável Z = NOT X, Z comportar-se-á como uma variável lógica, assumindo unicamente os valores lógicos 0 (falso) ou 1 (verdadeiro).

Como utilização de variáveis lógicas no TK, podemos efetuar o seguinte "macete":

Substituição do comando: IF w = 0 THEN ... por um comando de variável lógica:

O comando IF dos TKs verificam os níveis lógicos das variáveis da comparação e, através deles, estabelecem a validade ou não da expressão relacional e sua respectiva execussão, ou não.

Baseando-nos neste fato, podemos, em vez de fornecermos uma comparação ao comando IF, fornecermos diretamente o nível lógico da variável de comparação. Assim, podemos substituir:

```
100 IF W=0 THEN ...
```

por:

```
100 IF NOT W THEN ...
```

Como vantagem da substituição temos uma maior rapidez de cálculo e uma economia de 7 bytes de memória.

Equações Lógico-Aritméticas Gerais

Vemos, então, que as expressões matemáticas mais gerais que um TK pode processar é uma combinação de variáveis numéricas, alfanuméricas e lógicas, executadas por operadores aritméticos e lógicos.

Por exemplo, o programa abaixo calcula uma variável a partir de uma expressão aritmético-lógica:

```
  9 LET Z=INT (RND*100+1)
 10 LET F=SGN ((-3)*(RND>=0.5)+
3*(RND>0.8)-3+6*(NOT Z)-4*NOT (Z
<0 AND Z>10))
 11 PRINT F
```

Baseando-se nestas expressões lógico-aritméticas, podemos desenvolver diversos "macetes" de programação com a finalidade de otimizar a programação e economizar memória, e ainda aumentar o "poder" de nosso microcomputador. Por exemplo:

Substituição de um conjunto de IF's em cadeia por uma questão lógica

Consideremos que, em um programa, temos:

```
100 IF Y=0 THEN LET Z=Z+A
110 IF (Y>0) AND (Y<1) THEN LET
Z=3*Z+1
120 IF X<>L THEN LET Z=Z-1
130 IF COS (L)=PI/L THEN LET Z=
0
```

tal conjunto de instruções pode ser substituido pela seguinte equação lógica:

```
100 LET Z=VAL "Z+A*(NOT Y)+(3*Z
+1)*(Y>0 AND Y<1)-(X<>L)-Z*(COS
(L)=PI/L)"
```

A economia de memória é de 42 bytes. Poderiamos ainda utilizar a função VAL e obter.

```
10 LET Z=VAL "Z+A*(NOT Y)+(3*Z
+1)*(Y>0 AND Y<1)-(X<>L)-Z*(COS
(L)=PI/L)"
```

Neste caso a economia de memória é de 63 bytes.

Substituição de equações de movimentação em jogos por equação lógica:

Freqüentemente encontramos linhas de programação em jogos, do tipo:

```
100 IF INKEY$="5" THEN LET Y=Y-
1
110 IF INKEY$="8" THEN LET Y=Y+
1
```

linhas estas utilizadas para a movimentação de elementos na tela. Podemos substituir conjuntos de linhas deste tipo por equações lógicas do tipo:

```
100 LET Y=Y+(INKEY$="8")-(INKEY
$="5")
```

Como vantagem temos um aumento na velocidade de processamento e uma economia de 23 bytes.

Implementação dos comandos de desvio condicionado: ON k GOTO x, y, z, e ON k GOSUB x, y, z.

Tinhamos estabelecido um comando para o desvio condicionado, entretanto, tal comando apresentava a desvantagem de que: quando k, vindo do programa, não pertencia a faixa dos inteiros previstos pelo desvio, obtinhamos um comando do tipo GOTO 0.

Baseando-nos nas expressões lógico-aritméticas gerais, podemos contornar este problema. Assim proponho que:

```
NNN    ON K GOTO X,Y,Z,...
```

sejam substituidos por:

```
NNN GOTO X*(K=1)+Y*(K=2)+Z*(K=3)
+...+(NNN+1)*(K<1 OR K>F)
```

e:

```
NNN  GOSUB  X*(K=1)+Y*(K=2)+Z*(K
=3)+...+(NNN+1)*(K<1 OR K>F)
```

onde: NNN é o número da linha; X, Y, Z, . . .Z são as linhas de desvio e *f* é o último valor que K assume.

Por exemplo:

```
100 ON GOTO 30,11,70
```

fica como:

```
100 GOTO 30*(K=1)+11*(K=1)+70*(
K=3)+101*(K>1 OR K>3)
```

Claramente o último condicional lógico nos garante que: se K não estiver no intervalo de 1 a 3, então a equação lógica resume-se a 100 GOTO 101 e o programa prossegue normalmente.

A fim de economizarmos mais memória, é conveniente reescrever a equação lógica com VAL:

```
100 GOTO VAL "30*(K=1)+11*(K=1)
+70*(K=3)+101*(K>1 OR K>3)"
```

Generalização dos desvios condicionais para os TKs

A maioria dos computadores utilizam somente desvios condicionais controlados com variáveis inteiras. Os TKs apresentam uma grande vantagem: os controladores dos desvios são definidos, logicamente, pelos usuários. Assim podemos utilizar para controle dos desvios números reais (de ponto flutuante) e até mesmo expressões "Lógico-aritméticas".

Assim, se X é uma variável de ponto flutuante, que vem do programa para controlar o desvio, podemos ter, por exemplo:

a — Se: X = 0, desvio para linha 3
b — Se: X entre 0 e 5, desvio para linha 20
c — Se: X entre 5 e 10, desvio para linha 15
d — Se: X entre 10 e 20, desvio para linha 50
e — Considere-se que X não assume os valores: 5; 10.

Nestas condições podemos escrever a seguinte "Equação de Desvio Condicionado":

```
100 GOTO VAL "3*(NOT X)+20*(X>0
 AND X<5)+15*(X>5) AND X<10)+50*
(X>10 AND X<20)+101*(X<0 OR X>=2
0)"
```

Como exemplo de um "Programa Lógico", consideremos o seguinte programa, que determina quantas vezes a função RND tomou valores maiores e menores que 0.5 em um lote de amostragem de 10000 números:

```
 10 REM -EXEMPLO DE PROGRAMA LO
GICO
 20 REM MARIO CESAR S.BARBOSA
 30 LET F=0
 40 LET Y=0
 50 LET N=0
 60 LET X=RND
 70 LET N=N+1
 80 LET F=F+(X>=0.5)
 90 LET Y=Y+(X<=0.5)
100 IF NOT (N-10000) THEN GOTO
120
110 GOTO 60
120 SLOW
130 PRINT "F= ";F;"RND>=0.5"
140 PRINT "Y= ";Y,"RND<=0.5"
150 STOP
```

Sugestões aos leitores

Como sugestões aos leitores proponho uma análise e pesquisa das leis de formação das operações lógicas alfanuméricas e numérico-alfanuméricas tipo: "A" < "B"; "1" = "G"; "AB" > "AC".

Observe que é possível utilizar-se nos TKs desvios condicionados, controlados por Variáveis Alfanuméricas e até mesmo Números-alfanuméricos.

Como sugestão também pesquisem a utilização de álgebra lógica em "macetes de programação".

Você acabou de comprar o seu computador, compatível com o TK 83, retirou-o da caixa, instalou-o na TV, ligou sua fonte, apareceu o K invertido no canto inferior do vídeo, exatamente como dizia o manual.
— Muito bem, — pensa você — mas . . .

E agora?

Roberto Bertini Renzetti

Cronômetro de Precisão

Nesta edição apresentamos um curto programa que transforma seu computador num cronômetro digital de alta precisão!

Ele pode parecer um pouco complexo mas, na realidade, seu funcionamento é simples de ser entendido e não apresenta problemas de digitação.

Digitação

Linha 1Ø

O comando IF encontra-se na tecla U e serve para tomar decisões. Toda linha em Basic que tiver um IF, obrigatoriamente deverá ter um THEN (SHIFT + Tecla 3). INKEY$ é obtido com SHIFT + NEW LINE + Tecla B e tem a função de permitir a entrada de apenas um dado, através do teclado.

A linha que contém o INKEY$ fica esperando você pressionar alguma tecla para então passar para a linha seguinte. Caso contrário, ela "pula" para seu próprio começo e continua esperando.

Linhas 2Ø e 3Ø

O comando POKE tem a finalidade de introduzir um número em um determinado espaço (endereço) da memória.

No nosso programa, estes endereços representam duas variáveis do programa interpretador da ROM que servem para contar os "quadros" da tela de TV (veja o apêndice do seu manual). Esses endereços vão ser preenchidos com os números 255 e 127 respectivamente. Note que 255 + 256 * 127 = 32767.

Não é importante você saber mais detalhes por enquanto. Basta saber digitar o programa e ver como ele funciona.

Linha 4Ø

Esta "flechinha" é simples de ser obtida: primeiro digite SHIFT L e depois SHIFT M. Um aviso: esta "flecha" só serve para "enfeitar" na tela ou como desenho num joguinho — pois aparece entre aspas com um PRINT — e não tem a mesma função de "igual e maior" que existe em SHIFT Y.

Linhas 5Ø e 6Ø

Têm quase a mesma função que a linha 1Ø.

Linha 7Ø

O comando LET já foi explicado no "E Agora" da revista número 10. Já a função PEEK serve para "ler" um endereço da memória. Fazendo uma analogia: imagine as memórias como caixas, Com POKE, você *põe* alguma coisa dentro da caixa e com o PEEK você apenas *olha* e *copia* "a coisa" que está lá dentro (sem modificar nada).

Linha 8Ø

Vai imprimir o tempo gasto (quando o programa for rodado) entre apertar uma tecla uma vez e apertá-la de novo (como um cronômetro).

Linhas 9Ø e 1ØØ

Novamente, usamos INKEY$ só que, desta vez, para rodar o programa novamente.

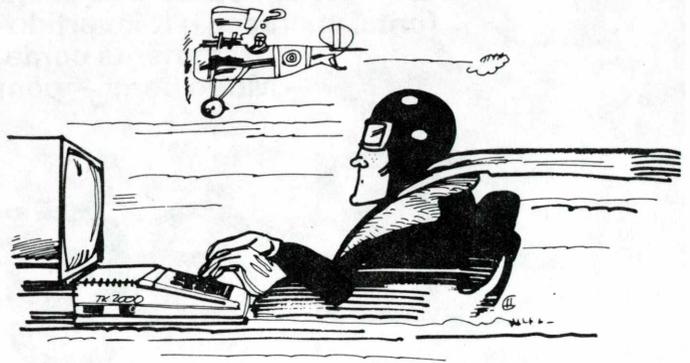
Como funciona o programa:

Ao rodar RUN, o programa espera que você digite uma tecla qualquer. Quando isto acontecer, aparecerá a flecha na tela e o "cronômetro" vai ser disparado. O tempo é marcado de acordo com a relação existente entre a freqüência da "varredura" da tela com as variáveis dos endereços 16436 e 16437.

O resultado (tempo) é brecado e apresentado na tela quando se apertar a tecla outra vez.

```
  10 IF INKEY$="" THEN GOTO 10
  20 POKE 16436,255
  30 POKE 16437,127
  40 PRINT "=>";
  50 IF INKEY$<>"" THEN GOTO 50
  60 IF INKEY$="" THEN GOTO 60
  70 LET A=PEEK 16436+PEEK 16437
*256
  80 PRINT (32767-A)/60;" SEGUND
OS"
  90 IF INKEY$<>"" THEN GOTO 90
 100 RUN
```

Arquivos

Luis Carlos Nunes
Wilson J. Tucci

Existem dois tipos de arquivo BASIC: os arquivos seqüenciais e os arquivos diretos (ou RANDOM). A principal diferença entre os dois é que nos seqüenciais, tanto a leitura quanto a gravação deve ser feita do início ao fim do arquivo, de registro a registro. O tamanho dos registros pode ser variável. Nos arquivos diretos, o acesso (Leitura/gravação) é feito diretamente para o registro que nos interessa. A única limitação é que os registros devem ter tamanho fixo. (figura 1)

Sejam os seguintes registros:
ARVORE
GADO
CACHORRO
GALINHA
UVA
Em um arquivo sequencial poderíamos ter:

|A|R|V|O|R|E|↵|G|A|D|O|↵|C|A|C|H|O|R|R|O|↵|G|A|L|I|N|H|A|↵|U|V|A|↵|

Mas em um arquivo direto deveríamos ter:

|A|R|V|O|R|E| | |↵|G|A|D|O| | | | |↵|C|A|C|H|O|R|R|O|↵|G|A|L|I|N|H|A| |↵|

|U|V|A| | | | | |↵|

ou seja, todos os registros deverão ter, pelo menos, 8 posições.

OBS: o símbolo ↵ representa "RETURN"

Fig. 1

Se quisermos acessar a palavra GALINHA no arquivo seqüencial devemos ler os 3 primeiros registros e só então poderemos acessar a palavra GALINHA. No arquivo direto, podemos acessar diretamente o 4º registro.

```
10 REM ACESSO SEQUENCIAL
20 D$=CHR$(4)
30 HOME
40 PRINT D$; "OPEN TESTE"
50 PRINT D$; "WRITE TESTE"
60 PRINT "ARVORE"
70 PRINT "GADO"
80 PRINT "CACHORRO"
90 PRINT "GALINHA"
100 PRINT "UVA"
110 PRINT D$; "CLOSE TESTE"
120 PRINT D$; "OPEN TESTE"
130 PRINT D$; "READ TESTE"
140 FOR I=1 TO 5
150 INPUT A$
160 NEXT
170 PRINT D$; "CLOSE TESTE"
180 END
```

Arquivos seqüenciais

Observe que na linha 20 nós fizemos D$ = CHR$ (4). Isto foi feito para que tenhamos acesso ao TKDOS. Necessitamos enviar ao computador (Sistema Operacional), um caractere de controle (CHR$(4)). Isto é feito através do comando PRINT.

Para iniciarmos o trabalho com um arquivo é necessário "abri-lo". Isto é feito com o comando OPEN. Ao terminarmos o trabalho, devemos "fechar" esse arquivo. O comando para isto é CLOSE. Observe que os comandos estão sempre entre aspas, precedidos pelo comando PRINT D$.

Devemos, depois, fornecer ao computador qual a operação que queremos fazer com esse arquivo (ler ou gravar). No caso, primeiramente, desejamos gravar neste arquivo. Assim devemos dar o comando WRITE.

A seguir, utilizamos o comando PRINT para gravar esses registros no arquivo teste. Como nós demos imediatamente antes o comando PRINT D$ — "WRITE TESTE" —, o microcomputador "sabe" que todos os comandos PRINT que se seguem, referem-se a "impressões" no disco e não no vídeo. Ao terminar o processo, damos o comando PRINT D$ — "CLOSE TESTE".

Neste instante, o comando PRINT volta a atuar no vídeo.

Logo depois, no programa, abrimos novamente o arquivo TESTE e lemos os cinco registros. Ler um registro, significa entrar com um valor. Por isso nos utilizamos o comando IMPUT (este, pelos mesmos motivos anteriores, atua agora no disco ao invés de atuar no teclado).

Ao finalizar o trabalho, digitamos "CLOSE TESTE".

Execute o programa. O que aconteceu?

A luz vermelha do drive acendeu e apagou, mas nada surgiu na tela. Por que? Porque nós não mandamos sair nenhum resultado na tela. Se você desejar saber se o programa está funcionando, basta digitar antes do comando RUN. Agora você verá as operações sendo executadas.

Seja agora um arquivo direto:

As maiores diferenças estão no comando OPEN; devemos declarar o tamanho dos registros e, tanto na gravação quanto na leitura, devemos especificar qual registro nós vamos acessar (R1, R2, R3, R4, R5).

Execute o programa colocando ou não o comando MON C, I, O e tente explicar o resultado.

Para "desligar" o comando MON C, I, O, basta digitar o comando NOMON C, I, O.

A seguir vamos propor um arquivo telefônico. Apresentaremos sua estrutura e dados necessários. No próximo número apresentaremos seu fluxograma e sua codificação.

```
10 REM ARQUIVO DIRETO
20 D$= CHR$(4)
30 PRINT D$; "OPEN TESTE, L8"
40 PRINT D$; "WRITE TESTE, R1"
50 PRINT "ARVORE"
60 PRINT D$; "WRITE TESTE, R3"
70 PRINT "GADO"
80 PRINT ;"WRITE TESTE, R3"
90 PRINT "CACHORRO"
100 PRINT D$; "WRITE TESTE, R4"
110 PRINT "GALINHA"
```

Agenda Telefônica

Os dados necessários para uma agenda telefônica são:
Nome: campo com 30 caracteres
Endereço: campo com 30 caracteres
CEP: campo com 05 caracteres
Telefone: campo com 13 caracteres
(. . .)

Vamos utilizar um arquivo direto (RANDOM) para armazenar esses registros.

N. registro

NOME	ENDEREÇO	CEP	TELEFONE

Um arquivo direto é acessado pelo seu número de registro. Vamos armazenar em um arquivo seqüencial, o nome e o número do registro.

NOME	N. DO REGISTRO

A idéia é colocar o nome e o número do registro do arquivo direto no seqüencial, em ordem alfabética, para auxiliar uma busca.
Exemplo:

Sejam os nomes a colocar (nesta ordem)

JOÃO, AV MIRUNA 322, CEP 04084 TEL 5212154
ANTONIO, AV SABIA 422, CEP 04515 TEL 714620
LUIZ, RUA CAMÕES 12 APTO 23, CEP 08200 TEL 202341

O arquivo direto ficará como mostramos na figura 2 e o sequencial ficará como mostramos na figura 3

A idéia é colocar o arquivo seqüencial em um vetor da memória, ordená-lo e usar novamente o arquivo seqüencial para isso.

Utilizaremos o método de classificação já visto ("jogador de cartas").

Assim para procurarmos um nome, devemos carregar o arquivo seqüencial na memória (vetor), localizá-lo, obter seu número de registro e ir diretamente ao arquivo "RANDOM" com esse número de registro.

Desse modo, utilizaremos menos a memória. (Imagine ter que carregar vetores na memória para armazenar o arquivo direto, dado o seu tamanho.)

As listagens desta seção foram obtidas na impressora Mônica da Elebra Informática.

A unidade de disco utilizada como periférico do TK 2000 é a Horacio da Elebra Informática.

1 JOAO	AV MIRUNA 322	( ) 5212154
2 ANTONIO	AV SABIA 422	( ) 714620
3 LUIZ	RUA CAMOES 12 APTO 23 43201	( ) 202341

Fig. 2

JOAO	1
ANTONIO	2
LUIZ	3

Fig. 3

ANTONIO	2
JOAO	1
LUIZ	3

Fig. 4

```
10 REM ACESSO SEQUENCIAL
20 D$ = CHR$ (4)
30 HOME
40 PRINT D$;"OPEN TESTE"
50 PRINT D$;"WRITE TESTE"
60 PRINT "ARVORE"
70 PRINT "GADO"
80 PRINT "CACHORRO"
90 PRINT "GALINHA"
100 PRINT "UVA"
110 PRINT D$;"CLOSE TESTE"
120 PRINT D$;"OPEN TESTE"
130 PRINT D$;"READ TESTE"
140 FOR I = 1 TO 5
150 INPUT A$
160 NEXT
170 PRINT D$;"CLOSE TESTE"
180 END
```

```
10 REM ARQUIVO DIRETO
20 D$ = CHR$ (4)
30 PRINT D$;"OPEN TESTE, L8"
40 PRINT D$;"WRITE TESTE, R1"
50 PRINT "ARVORE"
60 PRINT D$;"WRITE TESTE,R3"
70 PRINT "GADO"
80 PRINT D$;"WRITE TESTE,R3"
90 PRINT "CACHORRO"
100 PRINT D$;"WRITE TESTE, R4"
110 PRINT "GALINHA"
120 PRINT D$;"WRITE TESTE, R5"
130 PRINT "UVA"
140 PRINT D$;"CLOSE TESTE"
150 PRINT D$;"OPEN TESTE,L8"
160 PRINT D$;"READ TESTE,R4"
170 INPUT A$
180 PRINT D$;"CLOSE TESTE"
190 PRINT A$
200 END
```

Como localizar Cometas com o Apple ou TK 2000

Antonio Fernando Bertachini

Com o programa abaixo você poderá achar a posição de quinze cometas em qualquer data desejada.

I) Pequena introdução teórica

Além do Sol, dos planetas e de pequenos corpos chamados asteróides, o Sistema Solar possui alguns elementos de massa que são chamados cometas. Os cometas são pequenos corpos que descrevem órbitas fortemente elípticas em torno do Sol (isto é, sua distância até o Sol varia muito de um ponto a outro de sua órbita) e com período de revolução (tempo gasto para uma volta completa em torno do Sol) muito grande, da ordem de séculos em alguns casos.

Esses pequenos corpos são invisíveis até aos mais potentes telescópios do mundo durante a maior parte do tempo, só mostrando sua existência quando se aproximam do periélio (ponto da órbita onde ele fica mais próximo do Sol). É o que se chama de passagem ou chegada do cometa, e acontecerá no início de 1986 para o mais famoso dos cometas: o de Halley.

Quando essa aproximação do Sol ocorre, o cometa, que está carregado de gelo e poeira, se aquece e começa a perder matéria. Essa matéria é arrastada pelo vento solar (um fluxo de partícula que emana do Sol), e forma uma longa faixa brilhante denominada cauda do cometa. Note-se que, como essa cauda é formada pelo vento solar, ela sempre aponta na direção do Sol, onde quer que esteja o cometa.

Esses corpos brilhantes (em alguns casos) e caudalosos foram objetos de temor em séculos passados. Eram considerados como um aviso dos céus de que alguma desgraça em breve cairia sobre a humanidade. Esses preconceitos duraram por muito tempo, até que investigações de Halley, Apianus, Kepler e outros mostraram que os cometas são simples corpos celestes.

II) Como Localizar os Cometas

Os cometas periódicos (aqueles que passam pelo periélio em períodos regulares de tempo), podem ter suas posições calculadas a partir da Mecânica Celeste. Para isso é necessário conhecer os elementos orbitais (uma série de valores que especificam a posição de um corpo celeste) para uma determinada época (para os cometas a mais comum é a época da passagem pelo periélio). Com esses elementos, e o tempo decorrido entre a data base e a data desejada, calcula-se quanto o comenta deslocou-se e, portanto, sua nova posição.

Repetindo esses cálculos para a Terra, tem-se a posição da mesma e basta executar uma transformação de coordenadas para obter a posição do cometa no céu da Terra.

III) O Programa

O programa escrito em BASIC foi feito no microcomputador Unitron (versão nacional do Apple II), mas pode ser facilmente adaptado a outras versões de BASIC.

O programa principal começa apresentando um menu dos cometas para escolha do usuário. Feita essa escolha, o programa chama a sub-rotina DADOS (linha 2000) e esta carrega as variáveis PE, WB, OM, TP, AM, EM e IO com os valores da época do periélio (em anos decimais), longitude do periélio (graus), longitude do modo ascendente (graus), período, de revolução (anos), semi-eixo maior da órbita (U.A.), excentricidade da órbita (admencional) e inclinação da órbita (graus), respectivamente. Também os dados relativos à Terra (período orbital), longitude do periélio, longitude e excentricidade da órbita na data base de primeiro de janeiro de 1980 são carregados nas variáveis TE, WT, EP, e ET, respectivamente. Com as variáveis carregadas, a sub-rotina termina e é feita a volta ao programa principal.

A seguir o programa pede a data para a qual é desejada as informações e chama a sub-rotina ÉPOCA. Essa sub-rotina, que começa na linha 3000, transforma a data desejada em frações decimais de ano. Por exemplo, 15 de agosto de 1984, se transforma em 1984, 62423, já que os dias decorridos no ano de 1984 até 15 de agosto (288 dias), correspondem a 62,423% de um ano (que dura aproximadamente 365,25 dias).

Após esse cálculo é feita a volta ao programa principal, que calcula o valor de Y (diferença entre a época do periélio e a desejada, em anos decimais) e M (M = 360Y/TP, onde TP é o período de revolução do cometa, em anos). O valor de M é reduzido ao intervalo 0-360 graus e transformado em radianos. Com o valor de M e com EM (excentricidade da órbita do cometa), vindo da sub-rotina DADOS, o programa chama a sub-rotina SOLVE (linha 4000), que tem a tarefa de resolver a equação de Kepler, isto é, encontrar E tal que E-EM. sin (E) = M. Isso é feito adotando-se E — M, como tentativa inicial, e calculando a diferença D9 (D9 = E — EM. sin (E) — M). O processo iterativo continua, tomando o próximo valor de E como sendo o E anterior menos D8/(D8 = D9/(1 — EM.cos(E)), até que o valor absoluto de D9 seja menor que 0.000001.

De posse do valor de E, calcula-se a anomalia verdadeira V da fórmula tan (V/2) = (1 + EM)/(1 — EM). tan (E/2) e transforma-se o valor deste de radianos para graus.

Com o valor de V calcula-se a longitude heliocêntrica LC (LC = V + WB) e a distância do Sol RC (RC = AM. (1 — EM.EM)/(1 + EM.cos(V + 3,1416/180))), para que sejam projetados no plano da eclíptica e resultem em LL e RL. O valor da latitude heliocêntrica PSI é tirado da fórmula (1).

(1) PSI = arc tan(sin(LC — OM). 3,1416/180). sin(IO. 3,1416/180))

Agora que o cometa está localizado, precisamos localizar a Terra. Começamos por chamar a sub-rotina DIAS (que começa na linha 6000) e calcular o número de dias passados entre a data base (1º de janeiro de 1980) e data desejada. Com este valor o programa principal calcula NE(NE = 360.D/(TE./365,2422, onde TE é o período orbital da Terra) e ME (ME = NE + EP — WT), com EP e WT sendo a longitude e a longitude do período da Terra, respectivamente, na data base. Agora, calcula-se a longitude heliocêntrica da Terra L (L = NE + 360). ET + sin(ME. 3,1416/180)/3.1416 + EP), onde ET é a excentricidade da órbita da Terra e a distância ao Sol RT (RT = (1 — ET.ET)/(1 + ET.cos(VE 3,1416/180)), onde VE = L — WT é a anomalia verdadeira da Terra.

Com a Terra e o Cometa posicionados, ainda nos falta obter a posição em que este se encontra nos céus da Terra. Para isso, dividimos os esforços em duas partes: primeiro calcularemos sua posição no sistema eclíptico, especificando suas coordenadas LAM (sua distância, medida perpendicular-

mente à eclíptica, até a eclíptica) através das fórmulas:

(2) LAM = 180 + L + arc tan (RL.sin(L−LL)/(RT−RL.cos(L−LL))), se RL for maior que RT, isto é, se a projeção do cometa sobre a eclíptica se encontra mais perto do Sol do que a Terra, ou

(3) LAM = arc tan (RT.sin(LL−L)/(RL−RT.cos(LL−L))) + LL, em caso contrário.

(4) BET = arc tan (RL.tan(PSI).sin(LAM−LL)/(RT.sin(LL-L)))

Em segundo lugar usaremos a sub-rotina TRANSF (que começa na linha 5000), para mudar o sistema de coordenadas para o sistema equatorial. Nesse sistema são necessários as coordenadas da declinação (ângulo entre o equador celeste e o cometa) e a ascenção reta (mede a distância do primeiro ponto de Áries até o cometa, em horas).

Para finalizar os cálculos, usa-se a lei dos cossenos para calcular a que distância RO o cometa se encontra da Terra, ou seja:

RO = (RC.RC + RT.RT − 2.RC.RT.cos(LC−L).3,1416/180))

Ao leitor interessado em teoria de mecânica celeste, ou aos que desejam entender melhor os termos e equações usados na elaboração deste programa, recomenda-se a leitura do livro "Mecânica Celeste", de Nelson de Luca, da Editora da Universidade do Paraná.

IV) Acionando o programa

Após ter digitado o programa, basta teclar RUN e RETURN que ele lhe oferecerá o menu para escolha do cometa desejado. Com o menu na tela, basta digitar o número de sua escolha e pressionar a tecla RETURN. Agora é pedida a data e você deve fornecê-la no formato indicado, isto é: dia, mês e ano separados por vírgulas e o mês sendo um número de 1 a 12 e tecla RETURN (1,1,1986 para ver o Halley em primeiro de janeiro de 1986). Seu micro executará o programa e mostrará na tela os seguintes valores:

a) Cometa: Halley
b) Data: 1/1/1986
c) Declinação: −2 graus 08 min
d) Ascenção Reta: 22 horas 20 min
e) Distância da Terra: 0.212530
f) Distância do Sol: 1.027949

Com esse exemplo, você pode verificar se seu programa está correto. Note que esses valores foram obtidos com uma calculadora e talvez não estejam iguais aos seus nas últimas casas. Isso é devido à diferença de números com que as máquinas trabalham internamente.

Agora é só rodar o programa e procurar os cometas no céu, de preferência com um bom telescópio.

Nota: Antes de digitar este programa no TK 2000, digite MP, seguido por RETURN, pois ele ocupa mais de 6 kBytes.

Antonio Fernando Bertachini é acessor técnico da Kaxon Informática.

```
10  REM  COMETAS
20  REM  ANTONIO FERNANDO BERTACHINI
30  HOME
40  PRINT "ESCOLHA O COMETA:": PRINT
45  DIM C$(15)
50  FOR I = 1 TO 15
60  READ C$(I)
70  NEXT I
80  DATA  ENCKE,TEMPLE 2,HANEDA-CAMPOS
,SCHWASSMANN-WACHMANN 2,BORRELLY,
WHIPPLE,OTERMA,SCHAUMASSE,COMAS S
OLA,SCHWASSMANN-WACHMANN 1,NEUJMI
N 1,CROMMELIN,OLBERS,PONS-BROOKS,
HALLEY
90  FOR I = 1 TO 15
100  PRINT I;"- ";C$(I)
110  NEXT I
120  INPUT A
130  GOSUB 2000
140  HOME : PRINT "DATA (DD,MM,AAAA):"

150  INPUT DIA,MS,YR
160  GOSUB 3000
170 Y = DA - PE
180 MC = 360 * Y / TP
190  IF MC > 0 THEN 210
200 MC = MC + 360: GOTO 190
210  IF MC < 360 THEN 230
220 MC = MC - 360: GOTO 210
230 MC = MC * 3.1416 / 180
240  GOSUB 4000
250 V =  SQR ((1 + EM) / (1 - EM)) *  TA
N (E / 2)
260 V =  ATN (V) * 2:V = V * 180 / 3.1
416
270 LC = V + WB
280 RC = AM * (1 - EM * EM) / (1 + EM *
 COS (V * 3.1416 / 180))
290 PSI =  SIN ((LC - OM) * 3.1416 / 1
80) *  SIN (IO * 3.1416 / 180)
300 PSI =  ATN (PSI /  SQR (1 - PSI *
PSI))
310 Y =  SIN ((LC - OM) * 3.1416 / 180
) *  COS (IO * 3.1416 / 180)
320 X =  COS ((LC - OM) * 3.1416 / 180
)
330 AU =  ATN (Y / X)
340  IF X < 0 THEN AU = AU + 3.1416
350  IF Y < 0 AND X > 0 THEN AU = AU +
2 * 3.1416
355 AU = AU * 180 / 3.1416
360 LL = OM + AU
362  IF LL > 0 THEN 366
364 LL = LL + 360: GOTO 362
366  IF LL < 360 THEN 370
368 LL = LL - 360: GOTO 366
370 RL = RC *  COS (PSI)
380  REM  CALCULOS PARA A TERRA
390  GOSUB 6000
400 NE = 360 * D / TE / 365.2422
402  IF NE > 0 THEN 404
403 NE = NE + 360: GOTO 402
404  IF NE < 360 THEN 410
407 NF = NE - 360: GOTO 404
410 ME = NE + EP - WT
420 L = NE + 360 * ET *  SIN (ME * 3.1
416 / 180) / 3.1416 + EP
430  IF L < 360 THEN 450
440 L = L - 360: GOTO 430
450  IF L > 0 THEN 470
460 L = L + 360: GOTO 450
470 VE = L - WT
480 RT = (1 - ET * ET) / (1 + ET *  COS
(VE * 3.1416 / 180))
490  IF RL > RT THEN 530
500 AJ = RL *  SIN ((L - LL) * 3.1416 /
180) / (RT - RL *  COS ((L - LL) * 3.1
416 / 180))
510 LAM = 180 + L +  ATN (AJ) * 180 /
3.1416
520  GOTO 550
530 AJ = RT *  SIN ((LL - L) * 3.1416 /
180) / (RL - RT *  COS ((LL - L) * 3.1
416 / 180))
540 LAM = LL +  ATN (AJ) * 180 / 3.141
6
550  IF LAM > 0 THEN 570
560 LAM = LAM + 360: GOTO 550
570  IF LAM < 360 THEN 590
580 LAM = LAM - 360: GOTO 570
```

```
590 BET = RL * TAN (PSI) * SIN ((LAM
 - LL) * 3.1416 / 180) / RT / SIN ((LL
- L) * 3.1416 / 180)
600 BET = ATN (BET)
610 LAM = LAM * 3.1416 / 180
620 GOSUB 5000
630 RO = SQR (RC * RC + RT * RT - 2 *
RC * RT * COS ((LC - L) * 3.1416
 / 180))
640 HOME
650 PRINT "COMETA: ";C$(A)
660 PRINT
670 PRINT "DATA: ";DIA;"/";MS;"/";YR
680 PRINT
690 PRINT "DECLINACAO: ";
700 IF GA = 0 AND DO ( 0 THEN PRINT
"-";
710 PRINT GA;" GRAUS ";MIN;" MIN "
720 PRINT
730 PRINT "ASCENCAO RETA: ";AH;" HORA
S ";K5;" MIN"
740 PRINT
750 PRINT "DISTANCIA DA TERRA EM U.A.
: ";RO
760 PRINT
770 PRINT "DISTANCIA AO SOL EM U.A.:
";RC
780 END
2000 REM SUBROTINA QUE FORNECE OS DA
DOS
2010 DIM D(105)
2020 FOR I = 1 TO 105
2030 READ D(I)
2040 NEXT I
2050 DATA 1974.32,160.1,334.2,3.3,2.
209,.847,12,1972.87,310.2,119.3,5
.26,3.024,.549,12.5,1978.77,12.01
6,131.7,5.37,3.066,.64152,5.805
2060 DATA 1974.7,123.3,126,6.51,3.48
9,.386,3.7,1974.36.67.8,75.1.6.76
,3.576,.632,30.2,1970.77,18.2,188
.4,7.47,3.821,.351,10.2
2070 DATA 1958.44,150,155.1,7.88,3.9
58,.144,4,1960.29,138.1,86.2,8.18
,4.054..705,12,1969.83,102.9,62.8
,8.55,4.182,.577,13.4
2080 DATA 1974.12,334.1,319.6,15.03,
6.087,.105,9.7,1966.94,334,347.2,
17.93,6.858,.775,15,1956.82,86.4,
250.4,27.89,9.173,.919,28.9
2090 DATA 1956.46,150,85.4,69.47,16.
843,.93,44.6,1954.39,94.2,255.2,7
0.98,17.2,.955,74.2,1986.112,170.
011,58.154,76.0081,17.9435,.9673,
162.2385
2100 PE = D(7 * (A - 1) + 1)
2110 WB = D(7 * (A - 1) + 2)
2120 OM = D(7 * (A - 1) + 3)
2130 TP = D(7 * (A - 1) + 4)
2140 AM = D(7 * (A - 1) + 5)
2150 EM = D(7 * (A - 1) + 6)
2160 IO = D(7 * (A - 1) + 7)
2170 READ TE,EP,WT,ET
2180 DATA 1.00004,98.83354,102.59640
3,.016718
2190 RETURN
3000 REM SUBROTINA QUE CALCULA A DAT
A EM ANOS DECIMAIS
3010 DIM AU(12)
3020 FOR I = 1 TO 12
3030 READ AU(I)
3040 NEXT I
3050 DATA 0,31,59,90,120,151,181,212
,243,273,304,334
3060 DA = DIA + AU(M)
3070 IF (YR / 4 - INT (YR / 4) = 0) AN
D (MS ) 2) THEN DA = DA + 1
3080 DA = DA / 365.25 + YR
3090 RETURN
4000 REM SUBROTINA QUE RESOLVE A EQU
ACAO DE KEPLER
4010 E = MC
4020 D9 = E - EM * SIN (E) - MC
4030 IF ABS (D9) ( .000001 THEN 4070

4040 D8 = D9 / (1 - EM * COS (E))
4050 E = E - D8
4060 GOTO 4020
4070 RETURN
5000 REM SUBROTINA QUE TRANSFORMA CO
ORDENADAS ECLIPTICAS EM EQUATORIA
IS
5010 E1 = 23.441884 * 3.1416 / 180
5020 ANG = SIN (BET) * COS (E1) + COS
 (BET) * SIN (E1) * SIN (LAM)
5030 D1 = ATN (ANG / SQR (1 - ANG *
ANG))
5035 D1 = D1 * 180 / 3.1416
5040 DO = D1:D1 = ABS (D1)
5050 YS = SIN (LAM) * COS (E1) - TAN
(BET) * SIN (E1)
5055 AL = ATN (YS / COS (LAM)) * 180
 / 3.1416
5060 IF COS (LAM) ( 0 THEN AL = AL +
180
5070 IF YS ( 0 AND COS (LAM) ) 0 THEN
AL = AL + 360
5080 A1 = AL / 15
5090 FR = A1 - INT (A1)
5100 AH = INT (A1)
5110 K5 = INT (60 * FR)
5120 F1 = D1 - INT (D1)
5130 GA = INT (D1)
5140 GA = SGN (DO) * GA
5150 MIN = INT (60 * F1)
5160 RETURN
6000 REM SUBROTINA QUE CALCULA A DIF
ERENCA DE DIAS
6010 D7 = DS:M7 = MS:Y7 = YR
6020 IF M7 ( ) 1 AND M7 ( ) 2 THEN
6040
6030 Y7 = Y7 - 1:M7 = M7 + 12
6040 A7 = INT (Y7 / 100)
6050 B7 = 2 + INT (A7 / 4) - A7
6060 IF (YR ( 1582) OR (YR = 1582 AND
MS   10) OR (YR = 1582 AND MS = 1
0 AND DIA ( 15) THEN B = 0
6070 C7 = INT (365.25 * Y7)
6080 D8 = INT (30.6001 * (M7 + 1))
6090 JD = B7 + C7 + D8 + DIA + 1720994
.5
6100 D = JD - 2444238.5
6110 RETURN
```

Pesquisa de Raízes
Um exemplo de Recursão

Daniel R. Falconer
Wilson José Tucci

Temos abaixo um pequeno programa, em Pascal, que calcula a raiz quadrada de um número dado qualquer. Foi enviado pela nossa leitora Mônica Hoehne Mendes, de São Paulo, com a sugestão de que oferecêssemos um curso introdutório de Pascal, linguagem que atualmente ganha muito destaque nos Estados Unidos.

O programa que a Mônica mandou é um exemplo de uma ferramenta muito poderosa, e ao mesmo tempo interessante: a recursão. Sem entrar em maiores detalhes, uma função recursiva é aquela que, na sua definição, "recorre" a si mesma. Examinando a definição da função RAIZ temos, no ramo ELSE do IF, a atribuição ANTERIOR = RAIZ (X,N−1): a função RAIZ é definida em termos de si mesma.

O algoritmo empregado para encontrar a raiz quadrada de um número é o método de Newton, aplicado para encontrar um zero da função f(x) = X − a (onde a é o número cuja raiz quadrada queremos encontrar).

```
PROGRAM RAIZQUADRADA(INPUT,OUTPUT);

  CONST
    AVACANAOTOSSIR = TRUE;

  VAR
    A: REAL;
    N: INTEGER;

  FUNCTION RAIZ(X:REAL; N:INTEGER): REAL;
    (* Calcula a raiz quadrada de X
       recursivamente, pelo metodo de
       Newton, com N aproximacoes     *)

    VAR ANTERIOR: REAL;

  BEGIN
    IF N=Ø
       THEN RAIZ:= X/2
       ELSE BEGIN
              ANTERIOR:= RAIZ(X, N-1);
              RAIZ:= ANTERIOR -
         (ANTERIOR*ANTERIOR - X)/(2*X)
            END
    END;
BEGIN
  WHILE AVACANAOTOSSIR DO
    BEGIN
      READLN(A,N);
      WRITELN( RAIZ(A,N) )
    END
END.
```

TENTE ESTA

(para o TK 2000)

```
5  REM  CONTRATO DE RISCO
10  HGR
20  FOR A = 1 TO 7
30  HCOLOR  = A
40  FOR I = 0 TO 60 STEP A + 1
50  DRAW 10 AT 257,158: ROT= 2 * A:
SCALE= 200
70  NEXT I
80  NEXT A
90  GOTO 20
```

MÚSICA PARA O TK-2000

Vívian Bernardo

Saiba como aproveitar os recursos sonoros de seu micro.

O microcomputador TK-2000 pode reproduzir sons via alto-falante da TV ao qual está conectado. Isto permite o uso de sons em seus programas de modo a torná-los mais interessantes e criativos, prendendo a atenção do usuário com os efeitos audio-visuais. Para realizar esta função, o TK 2000 possui a instrução SOUND que gera som em um programa BASIC. O formato dessa instrução é SOUND X,Y onde X é a nota e Y é sua duração. Outro formato da instrução: SOUND X1, Y1 to X2, Y2 TO . . . Xn, Yn. Este formato reproduzirá uma série de notas seqüencialmente (uma escala de sons). A tonalidade das notas, em escala, pode variar de Ø a 255. Na tabela I, você encontra uma série de tons e seu valor correspondente.

Tabela I

SOL	255	128	64	31	15
LAb	243	121	6Ø	29	14
LA	231	114	56	28	13
SIb	217	108	53	26	12
SI	203	102	5Ø	25	
DO	192	96	47	23	
DO	182	90	45	22	
RE	172	85	42	21	
MIb	162	80	40	20	
MI	154	76	37	18	11
FA	146	72	35	17	10
FA	137	67	33	16	9

A duração das notas também pode variar de Ø a 255, sendo que uma mínima tem o valor de 24Ø, uma semínima tem valor de 12Ø, a colcheia tem valor 6Ø, semicolcheia 3Ø, fusa 15 e semifusa 8. Uma música pode conter notas pontuadas que geram tempo de compasso não inteiro; por isso são disponíveis 256 valores, de forma a permitir que você reproduza uma música com o máximo de precisão.

O número 1 é usado para pausa. Nos programas em BASIC, a forma mais usada para executar músicas é codificando as notas e pausas, de acordo com os códigos das tabelas, e colocá-las em matrizes. Também é possível obter efeitos sonoros mais complexos usando linguagem de máquina. Seu programa ganhará em velocidade. O canal de som é acessado a nível de linguagem de máquina com PEEK (— 16336) ou PEEK (49200) em BASIC. Cada acesso a esse endereço (no modo monitor = $CO30) produz transição de sinal que se repetido várias vezes com certa freqüência, gera som. Dessa forma, a diversidade de sons gerados é muito grande; é possível também reproduzir o som de acordes musicais (notas tocadas simultaneamente). Na verdade, os sons são emitidos distintamente; a rapidez de acesso ao canal de som em linguagem de máquina faz com que percebamos estes sons como emitidos ao mesmo tempo.

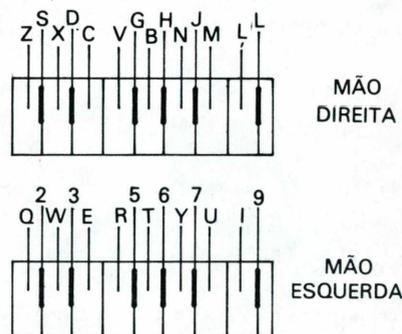
Foram elaborados quatro programas em BASIC para exemplificar o uso dos recursos de geração de som do TK-2000.

Programa 1

Esse programa executa a música "Tico-Tico no Fubá" (suas 110 primeiras notas). A estrutura do programa é bastante simples. As notas e as pausas são os dados armazenados nas linhas DATA do programa e estão, é claro, traduzidas para os valores entre 0 e 255 correspondentes. Após a leitura das matrizes, é executada a música (comando SOUND). A matriz nota e a matriz pausa têm a mesma dimensão porque cada nota tem uma duração. A carga das matrizes é feita na instrução 100 (comando READ). Os dados da matriz nota estão nas instruções 25 e 3Ø e os dados da matriz pausa, nas 52 e 53. A execução é feita com a instrução SOUND N(I), P(I) * 60. O fator multiplicativo 60 é colocado para evitar que nas linhas de DATA das pausas sejam digitados números grandes (as durações são todas múltiplas de 60). Quanto menor o fator, mais rápida é a execução. Caso a partitura exija reproduzir notas com durações não inteiras, você deve digitar a duração na linha de DATA, porque não poderá usar um fator múltiplo comum. Após digitar esse programinha e ver como ele funciona, escolha outra partitura, codifique-a e substitua as linhas DATA alterando a dimensão das matrizes e ouça sua música predileta no micro.

```
10  REM  PROGRAMA DE GERACAO DE MUSICA
 PARA O TK-2000
12  REM  TICO-TICO NO FUBA
15  DIM N(110),P(110)
25  DATA  37,42,47,50,56,60,50,47,42,3
7,37,76,80,76,72,76,56,76,80,76,7
2,76,60,76,80,76,72,76,42,50,60,7
6,85,90,96,56,60,64,72,56,42,47,5
0,56,76,56,47,47,50,53,50,102,80,
67,56,47,50,56
30  DATA  60,37,50,37,76,80,76,72,76,5
6,76,80,76,72,76,60,76,80,76,72,7
6,42,50,60,76,85,90,96,56,60,64,7
2,56,42,47,50,56,76,56,47,47,50,5
3,50,76,60,50,37,42,47,50,56
52  DATA  1,2,1,2,2,2,2,2,2,4,4,1,1,1,
1,1,2,1,1,1,1,1,2,1,1,1,1,1,1,1
,1,1,1,4,1,1,1,1,1,2,1,1,1,1,1,2,
1,1,1,1,1,1,1,1,1,1,1
53  DATA  1,2,1,1,1,1,1,1,1,2,1,1,1,1,
1,2,1,1,1,1,1,1,1,1,1,1,1,4,1,1,1
,1,1,2,1,1,1,1,1,2,1,1,1,1,1,1,1,
1,1,1,1,4
100  REM  LEITURA DA MATRIZ NOTA
105  FOR I = 1 TO 110: READ N(I): NEXT
I
110  REM  LEITURA DA MATRIZ PAUSA
115  FOR I = 1 TO 110: READ P(I): NEXT
I
120  REM  EXECUCAO
125  FOR I = 1 TO 110: SOUND N(I),P(I)
 * 60: NEXT I
130  END
```

Programa 2

Este programa foi elaborado para simular um teclado de piano no teclado do TK-2000. Como você sabe, o TK-2000 não possui a instrução INKEY$ que permite conectar o teclado diretamente com a máquina. Usamos outra técnica para simular essa função: a instrução PEEK (39) permite descobrir o código de cada tecla do micro. Com o código, basta usar o programa para introduzir o som que você quizer.

Teclado: A mão direita do piano (escala 4) está nas teclas de Z à L de acordo com o esquema I.; a mão esquerda (escala 2 do piano) está nas teclas de Q à 9 (figura 1).

O programa após o comando RUN, sonoriza o que você tocar, ou melhor, a tecla que você pressionar. A lógica de programação usada constitui em uma série de IFs encadeados, que comparam qual a tecla pressionada e emite o som à ela designado. A instrução 68 faz com que as comparações sejam feitas sempre a cada tecla digitada. A velocidade de comparação é relativamente grande. O programa permite que você toque qualquer música que tenha somente uma oitava em cada mão (não é possível tocar simultaneamente). Após o comando RUN, o programa só para ao digitar-se CONTROL C (BREAK). Caso você queira ampliar o teclado e colocar outras escalas de piano, basta seguir a seguinte rotina:

```
10  PRINT  PEEK (39): GOTO 10
```

Você obterá o número 42 que é o código da tecla RETURN. Em seguida aparecerá o número 48 que é o código indicando a inatividade do teclado. Pressionando qualquer tecla você obterá seu código. Tendo o código, você poderá acrescentar nas comparações do programa um som associado à tecla escolhida.

Por exemplo: você digita a tecla H, obtém o código 37, adiciona a seguinte comparação no programa:

```
100  IF  PEEK (39) = 37 THEN  SOUND X(
I),Y(I)
```

É fácil, não? Você pode executar as músicas que quizer e até compor com a ajuda de seu micro.

```
]LIST

10  REM  PROGRAMA DE GERACAO DE MUSICA
 NO TECLADO
20  REM  MICROMEGA- OUTUBRO 1984
30  REM  TECLADO
40  IF  PEEK (39) = 5 THEN  SOUND 96,1
20
41  IF  PEEK (39) = 10 THEN  SOUND 90,
120
42  IF  PEEK (39) = 4 THEN  SOUND 85,1
20
43  IF  PEEK (39) = 9 THEN  SOUND 80,1
20
44  IF  PEEK (39) = 3 THEN  SOUND 76,1
20
45  IF  PEEK (39) = 2 THEN  SOUND 72,1
20
46  IF  PEEK (39) = 7 THEN  SOUND 67,1
20
47  IF  PEEK (39) = 1 THEN  SOUND 64,1
20
48  IF  PEEK (39) = 37 THEN  SOUND 60,
120
49  IF  PEEK (39) = 43 THEN  SOUND 56,
120
50  IF  PEEK (39) = 38 THEN  SOUND 53,
120
51  IF  PEEK (39) = 44 THEN  SOUND 50,
120
52  IF  PEEK (39) = 45 THEN  SOUND 47,
120
53  IF  PEEK (39) = 40 THEN  SOUND 45,
120
54  IF  PEEK (39) = 17 THEN  SOUND 192
,120
55  IF  PEEK (39) = 22 THEN  SOUND 182
,120
56  IF  PEEK (39) = 16 THEN  SOUND 172
,120
57  IF  PEEK (39) = 21 THEN  SOUND 162
,120
58  IF  PEEK (39) = 15 THEN  SOUND 154
,120
59  IF  PEEK (39) = 14 THEN  SOUND 146
,120
60  IF  PEEK (39) = 19 THEN  SOUND 137
,120
61  IF  PEEK (39) = 13 THEN  SOUND 128
,120
62  IF  PEEK (39) = 25 THEN  SOUND 121
,120
63  IF  PEEK (39) = 31 THEN  SOUND 114
,120
64  IF  PEEK (39) = 26 THEN  SOUND 108
,120
65  IF  PEEK (39) = 32 THEN  SOUND 102
,120
66  IF  PEEK (39) = 33 THEN  SOUND 96,
120
67  IF  PEEK (39) = 28 THEN  SOUND 90,
120
68  GOTO 40
```

Programa 3

Esse programa vai ensiná-lo a usar o canal de som do micro, com acesso pela instrução POKE. É uma rotina que dispensa o comando SOUND. Digite a rotina:

```
10  REM  - ** ROTINA DE GERACAO DE SOM
 NO TK-2000 COM O USO DA INSTRUCA
O 'POKE' **
20  FOR L = 770 TO 790
30  READ V
40  POKE L,V
50  PRINT V
60  GET A$
70  NEXT L
80  DATA  173,48,192,136,208,5,206,1
90  DATA  3,240,9,202,208,245,174,0
100  DATA  3,76,2,3,96
999  END
```

Digite o comando RUN e execute a rotina. Você obterá os valores das instruções DATA. Em seguida, digite NEW para que essa rotina entre na memória do computador. Os endereços 770 à 790 contêm os códigos sonoros do micro, e a rotina residente a partir desse endereço pode ser chamada, dentro de um programa em BASIC, para gerar sons. O programa que utiliza essa rotina está a seguir.

Sua estrutura lê os dados e finaliza quando encontra o nº 0 na linha DATA (instrução 1020). Os dois endereços anteriores ao 770 são preenchidos com nota e duração através da instrução PO-

KE e, em seguida, a rotina é chamada (CALL 770). Você pode ver que este programa comporta qualquer tipo de música, sendo mantida a estrutura das instruções 1000 a 1060, modificando-se apenas as linhas DATA, de acordo com a música, ou melhor os códigos correspondentes aos tons e pausas da música. Para finalizar não esqueça de colocar zeros na linha DATA: Este programa, como você pôde ver, não usa o comando SOUND e pode, portanto, rodar no APPLE e compatíveis.

```
1000  REM  - ** PROGRAMA DE MUSICA UTI
LIZANDO A ROTINA DE SOM **
1010  READ L: READ P
1020  IF L = 0 THEN  END
1030  POKE 768,L
1040  POKE 769,P
1050  CALL 770
1060  GOTO 1010
1070  DATA  192,100,154,100,128,100,12
8,200,76,100,76,200,96,100,96,200
1080  DATA  192,100,192,100,154,100,12
8,100,128,200,72,100,72,200,102,1
00,102,200
1090  DATA  203,100,203,100,172,100,11
4,100,114,200,72,100,72,200,85,10
0,85,200
1100  DATA  203,100,203,100,172,100,11
4,100,114,200,76,100,76,200,96,10
0,96,200
1110  DATA  192,100,192,100,154,100,12
8,100,96,200,64,100,64,200,76,100
,76,200
1120  DATA  192,100,192,100,154,100,12
8,100,96,200,56,100,56,200,72,100
,72,200
1130  DATA  85,100,85,100,72,100,56,10
0,56,250,67,100,64,100,37,250,47,
100,76,100,76,200,85,100,56,200,6
4,100,96,250,96,100
1140  DATA  0,0
1150  END
```

Programa 4

Vamos mostrar agora como unir som e Movimento em um programa. Essa técnica pode ser muito utilizada na animação de jogos, ou em efeitos audiovisuais de um programa em BASIC. Neste programa, uma figura movimenta-se na tela e simultaneamente ouve-se música. O programa principal está entre as linhas 20 e 40 que apresentam um "loop" para o movimento da figura na tela. A rotina que começa na instrução 1000 é acionada na linha 25, constrói a figura e emite som a cada linha impressa na tela. Há uma particularidade quanto aos caracteres de impressão usados para a montagem da figura: são utilizados caracteres especiais de vários formatos diferentes. Os caracteres especiais são acionados digitando-se CONTROL B e as teclas SHIFT e CONTROL, juntamente com a letra ou tipo do teclado escolhido. Também, digitando-se somente SHIFT obtém-se outros caracteres do teclado. Olhe a listagem do programa 4 e você verá que nas linhas onde os caracteres especiais foram usados aparecem letras minúsculas e letras maiúsculas misturadas dentro das aspas. Isso aconteceu porque a impressora usada foi uma Mônica, da Elebra, com formatação padrão ANADEX, que não imprime os caracteres especiais em seu formato correto; imprime o código interpretado (as letras minúsculas e maiúsculas que você vê na listagem do programa).

Para digitar uma linha de impressão usando os caracteres especiais do TK-2000, você deve proceder da seguinte forma:
PRINT "digite CONTROL-B, digite os caracteres especiais que desejar (use CONTROL SHIFT + tecla ou SHIFT + tecla), digite novamente CONTROL-B".

Na tabela II, mostramos os caracteres especiais usados no programa. Digite o comando CONTROL B no seu TK-2000 e descubra você mesmo os outros caracteres especiais.

Tabela II

SHIFT CONTROL	SHIFT	CARACTERE ESPECIAL
1		—
	A	/
	S	\
	D	┌
	F	┐
	G	┴
G		=
D		—
C		ǀ
	C	└
V		ǀ
	V	┘
	B	┤
	N	├
	Z	\
	X	╯

```
2  REM  SOM E MOVIMENTO
5  DIM N(52),P(52)
10  HOME
12  FOR I = 1 TO 52: READ N(I): NEXT I
14  FOR I = 1 TO 52: READ P(I): NEXT I
15 I = 0: RESTORE
20  FOR C = 1 TO 8
22  IF C = 5 THEN  RESTORE :I = 0: FOR
J = 1 TO 200: NEXT J
25  GOSUB 1000
30  NEXT C
35  FOR J = 1 TO 300: NEXT J
40  GOTO 10
1000  LET R = 5
1010  VTAB R: HTAB C
1020  PRINT "             rl---rm   "
1025  LET I = I + 1: SOUND N(I),P(I) *
60
1030  VTAB R + 1: HTAB C
1035  PRINT "             rZ   r[   "
1038  LET I = I + 1: SOUND N(I),P(I) *
60
1040  VTAB R + 2: HTAB C
1045  PRINT "             rZ   r[   "
1047  LET I = I + 1: SOUND N(I),P(I) *
60
1050  VTAB R + 3: HTAB C
1055  PRINT "             rZ   r[   "
1057  LET I = I + 1: SOUND N(I),P(I) *
60
1060  VTAB R + 4: HTAB C
1065  PRINT "  rZrXr[     rZrArArAr[
r["
1067  LET I = I + 1: SOUND N(I),P(I) *
60
1070  VTAB R + 5: HTAB C
1075  PRINT "  rZ rXrXrXrXrXrX    rXr
XrXr["
1078  LET I = I + 1: SOUND N(I),P(I) *
60
1080  VTAB R + 6: HTAB C
1085  PRINT "  -rZrrrYrY        r_    rb"
1086  LET I = I + 1: SOUND N(I),P(I) *
60
1088  VTAB R + 7: HTAB C
1090  PRINT "  rZ rArArArArArA    rAr
ArAr["
1091 I = I + 1: SOUND N(I),P(I) * 60: VT
AB R + 8: HTAB C
1092  PRINT "  rZrAr[     rZrXrXrXr[
r["
1093 I = I + 1: SOUND N(I),P(I) * 60: VT
AB R + 9: HTAB C
1094  PRINT "             rZ   r[   "
1095 I = I + 1: SOUND N(I),P(I) * 60
1100  VTAB R + 10: HTAB C: PRINT "
        rZ   r[   "
1105 I = I + 1: SOUND N(I),P(I) * 60
1110  VTAB R + 11: HTAB C: PRINT "
        rZ   r[   "
1112 I = I + 1: SOUND N(I),P(I) * 60
1115  VTAB R + 12: HTAB C: PRINT "
        rn---ro   "
1116 I = I + 1: SOUND N(I),P(I) * 60
1117  RETURN
1120  DATA  47,42,35,37,42,31,31,31,28
,37,35,42,42,43,35,37,42,47,23,25
,28,31,35,37,42
1130  DATA  47,42,35,37,42,31,31,31,28
,37,35,42,42,42,35,37,42,47,31,42
,37,47,31,42,37,47
1135  DATA  47
1140  DATA  4,1,1,1,1,2,2,1,1,1,1,2,2,
1,1,1,1,1,1,1,1,1,1,1,1,4,1,1,1,1
,2,2,1,1,1,1,2,2,1,1,1,1,1,1,1,1,
1,1,1,1,4
1145  DATA  4
```

Após digitar todos esses programas de música, tente você mesmo fazer outros com músicas de sua preferência, de sua autoria, ou até mesmo jogos incrementados de sons iguais aos das máquinas de fliperama. Então mãos à obra. Use e abuse do som de seu micro!

Disco TK

Gustavo Egídio de Almeida

Este programa funciona basicamente como um arquivo onde, ao invés de arquivar fichas ou dados pessoais, são arquivados discos, ou seja, uma discoteca.

Para você que dispõe de diversos discos (que geralmente ficam empilhados relaxadamente em estandes empoeiradas e sujeitos a um "desmoronamento") e costuma mudá-los de lugar constantemente, não se importando muito em classificá-los de acordo com as músicas, grupos ou conjuntos musicais e que, com esta constante mudança de lugar, nunca sabe onde está aquele determinado disco a que tanto procura naquele momento para ouvir aquela tão desejada música, fizemos o "Disco TK" — um programa que arquiva seus discos, numerados e colocados na ordem que você quiser.

Para 16K de memória é aceito um máximo de 40 discos, quantidade essa que pode ser ampliada com acoplamento de uma expansão de memória de 48K ou 64K.

Descrição

Ao ser carregado, o programa sai "rodando" e aparece no vídeo o menu com todas as instruções de uso, cada uma contendo sua respectiva tecla de acionamento (figura 1).

```
M-APRESENTACAO DOS DISCOS.
L-LISTAGEM DO DISCO.
A-ALTERACAO
R-ACRESCIMO
N-NOVO ARQUIVO
G-GRAVACAO DO NOVO ARQUIVO
```

Fig. 1

A única tecla que você terá acesso neste momento, por medida de segurança, será a tecla N.

A explicação para tal fato é que, caso você pressionasse alguma outra tecla, por exemplo a tecla M, deveria ser apresentado no vídeo todos os títulos das músicas. Porém, como todos esses títulos são armazenados na matriz DIM A$ e neste momento ela está vazia, ocorrerá fatalmente um código de erro no canto inferior da tela, sendo o programa interrompido neste ponto. A tecla N serve para iniciar um arquivo. Ao ser pressionada será apresentada uma mensagem, solicitando o número de discos do arquivo. Neste momento, você deverá digitar um número entre 1 e 40 e pressionar NEW LINE logo após.

A seguir, você deverá escrever o título de cada disco, digitando NEW LINE após a inserção de cada um.

Junto a cada título, você poderá colocar também o nome do conjunto, orquestra, etc., assim como o ano em que foi lançado, lembrando que no máximo são aceitos 50 caracteres por título.

Após serem digitados todos os títulos, será pedido na tela que você entre com a listagem das músicas de todos os discos, lembrando que no máximo são aceitos 200 caracteres por listagem. Para cada listagem pedida é mostrado na tela o respectivo título do disco.

Após serem inseridas todas as listagens, surgirá na tela uma mensagem indicando que o arquivo foi registrado, surgindo o menu inicial logo em seguida.

Agora qualquer uma das teclas do menu poderão ser acessadas. Caso você pressione a letra M, surgirá na tela uma apresentação contendo todos os títulos das músicas.

Para obter a listagem dos discos, pressione L e será pedido o número correspondente ao disco. Ao digitar esse número, surgirá na tela o título do disco com sua respectiva listagem.

Se você desejar fazer alterações em algum título, acrescente algum dado (data, nome do cantor, conjunto . . .) ou substitua-o por algum outro título, bastando para isto, pressionar a tecla A. Será então pedido o número do disco a ser alterado, a seguir o novo título do disco e posteriormente, sua listagem.

Caso você venha a aumentar sua coleção de disco e queira aumentar ainda mais o seu arquivo, use a tecla R, que lhe será pedido um novo título e posteriormente a sua listagem.

Para finalizar as instruções do menu temos como última opção a gravação do arquivo (tecla G).

Será pedido o nome do novo arquivo e a seguir será efetuada a operação de gravação.

Se você dispuser de um micro com maior capacidade de memória e desejar uma ampliação do arquivo, serão necessárias algumas modificações que devem ser seguidas com atenção. Dê uma olhada na linha abaixo:

```
4█LET K=40
```

Nessa linha armazena-se na variável K, que corresponde ao número máximo de discos que o arquivo suporta. Possuindo um TK com maior capacidade de memória, podemos alterar essa linha com um valor mais alto para a variável K. Lembrando-se que para alterar esse valor, devemos alterar também o valor da linha 1003:

```
1003 IF I>40 THEN GOTO 1000
```

Para digitar o programa, devemos seguir a seguinte ordem:

1) Digite primeiramente o programa monitor da figura 2, lembrando que a linha 1 REM deve conter 21 caracteres quaisquer.

2) Tecle RUN (NEW LINE) e entre com os códigos em decimal listados na figura 3.

3) Após a entrada desses códigos, delete as linhas 9000 em diante.

4) Dê o comando direto POKE 16510,0 seguido de NEW LINE.

5) Complete seu programa com a listagem a seguir:

```
   1 REM 000000000000000000000
9000 FOR F=16514 TO 16514+20
9010 PRINT F;"...";
9020 INPUT I
9030 PRINT I
9040 POKE F,I
9050 NEXT F
```

Fig. 2

```
16514...237      16525...124
16515...91       16526...205
16516...12       16527...0
16517...64       16528...103
16518...42       16529...1
16519...12       16530...181
16520...64       16531...2
16521...125      16532...237
16522...198      16533...176
16523...33       16534...201
16524...111
```

Fig. 3

Errata:

Investigação Policial, publicado na edição 14.

Nilson Martelo

"Um leitor me telefona e chama a atenção para a sub rotina da página 18, "que não roda". Realmente não roda do jeito que está. Correção:

Na linha 1030, deveria ser:

1030 PRINT I; TAB 8;

(Faltou o ponto e vírgula após o 8)"

Agradecemos o autor do artigo por ter apresentado esta falha, que agora transmitimos aos leitores.

Listagem

```
  0■REM GOSUB ?£RNDE£RND?LEN 5?
?LN  ?"■" GOSUB ■TAN
  2 REM REGISTRO DE DISCOS
  3 SLOW
  4 LET K=40
  5 DIM A$(K,200)
  6 DIM B$(K,50)
  7 LET I=0
 20 LET X=1
100 CLS
102 POKE 16418,2
105 PRINT AT 0,10;"DISCOTK";AT
0,10;"■■■■■■■"
110 PRINT AT 7,0;"M-APRESENTACA
O DOS DISCOS."
120 PRINT AT 9,0;"L-LISTAGEM DO
DISCO."
130 PRINT AT 11,0;"A-ALTERACAO"
140 PRINT AT 13,0;"R-ACRESCIMO"
145 PRINT AT 15,0;"N-NOVO ARQUI
VO"
147 PRINT AT 17,0;"G-GRAVACAO D
O NOVO ARQUIVO"
150 IF INKEY$="M" AND I>0 THEN
GOTO 200
155 IF INKEY$="G" AND I>0 THEN
GOTO 2000
160 IF INKEY$="A" AND I>0 THEN
GOTO 1045
170 IF INKEY$="R" AND I>0 THEN
GOTO 950
180 IF INKEY$="L" AND I>0 THEN
GOTO 1500
185 IF INKEY$="N" THEN GOTO 990
190 GOTO 105
200 LET C=1
201 LET X=0
205 CLS
207 POKE 16418,0
210 PRINT AT 22,10;"■■■■■■■■"
220 LET X=X+1
225 IF X>I THEN LET X=1
226 IF NOT INKEY$="" THEN LET B
$(X)=B$(I)
230 PRINT AT 18,0;B$(X);"/// ";
X
235 FOR F=1 TO 20
236 NEXT F
245 LET O=USR 16514
246 LET O=USR 16514
247 LET O=USR 16514
260 IF NOT INKEY$="" THEN GOTO
100
270 GOTO 215
950 CLS
951 PRINT AT 10,2;"QUAL DISCO D
ESEJA ARQUIVAR"
958 LET X=I+1
959 PRINT AT 13,0;"DE O TITULO
"
960 INPUT B$(X)
965 PRINT AT 13,0;"DE A LISTAGE
M"
967 INPUT A$(X)
969 CLS
970 PRINT AT 10,2;"       DISCO A
RQUIVADO   "
972 FOR W=1 TO 40
973 NEXT W
975 LET I=I+1
980 GOTO 100
990 CLS
995 PRINT AT 0,0;"QUANTOS DISCO
S DESEJA ARQUIVAR"
997 PRINT ,,"(MAXIMO->40 DISCOS
)"
1000 INPUT I
1003 IF I>40 THEN GOTO 1000
1004 PRINT AT 0,0;"DIGITE OA TIT
ULOS DOS DISCOS"
1005 LET K=1
1006 PRINT ,,"CADA TITULO DEVE C
ONTER       50 LETRAS NO MAXIMO"
1009 FOR C=1 TO K
1010 INPUT B$(C)
1015 NEXT C
1019 PRINT AT 0,0;"DIGITE A LIST
AGEM DE CADA DISCO"
1020 PRINT ,,"CADA LISTAGEM DEVE
 CONTER         200 LETRAS NO MAX
IMO"
1021■FOR C=1 TO K
1022 PRINT AT 10,0;B$(C)
1024 INPUT A$(C)
1025 NEXT C
1029 CLS
1030 PRINT AT 0,0;"          ARQUIV
O REGISTRADO"
1035 FOR F=1 TO 50
1036 NEXT F
1040 GOTO 100
1045 CLS
1050 PRINT AT 5,0;"DIGITE O NUME
RO DO DISCO QUE IRA SER ALTERADO
"
1055 INPUT T
1056 CLS
1060 PRINT AT 5,0;"DIGITE O NOVO
 TITULO DO DISCO"
1070 INPUT B$(T)
1075 PRINT AT 5,0;"DIGITE A LIST
AGEM DO DISCO                    "
1077 INPUT A$(T)
1080 PRINT AT 5,0;"NOVO DISCO AR
QUIVADO"
1090 FOR F=1 TO 50
1091 NEXT F
1100 GOTO 100
1500 CLS
1505 PRINT AT 0,0;"DIGITE O NUME
RO CORRESPONDENTE AO DISCO E OBT
ENHA A LISTAGEM DAS MUSICAS."
1510 INPUT Z
1515 LET C=Z
1516 IF Z=0 THEN GOTO 100
1517 PRINT AT 5,0;"TITULO";AT 5,
0;B$(C)
1518 PRINT "--------------------
------------"
1520 PRINT ,,"LISTAGEM"
1521 PRINT
1522 PRINT A$(C);"/// ";C
1525 PRINT AT 21,0;"P/VOLTAR AO
MENU INICIAL TECLE 0"
1610 GOTO 1510
1615 STOP
1620 SAVE "A■"
1630 RUN
2000 CLS
2005 PRINT AT 10,0;"DE O NOME DO
 NOVO ARQUIVO"
2006 INPUT S$
2007 CLS
2010 PRINT AT 10,0;"COLOQUE O GR
AVADOR EM POSICAO   DE GRAVAR E
TECLE NEW LINE"
2020 INPUT A$
2030 SAVE S$
2040 PRINT "ARQUIVO GRAVADO"
2045 PRINT AT 20,0;"PRESSIONE NE
W LINE"
2050 IF INKEY$="" THEN GOTO 2050
2060 GOTO 100
```

Como Colaborar com Microhobby

Temos recebido várias cartas de pessoas interessadas em colaborar conosco, perguntando quais os critérios para publicação. Embora tenhamos usado, várias vezes, um anúncio da casa onde indicávamos como colaborar com Microhobby, resolvemos colocar neste número novas regras, de maneira a nos organizarmos melhor, garantindo tanto os interesses dos leitores, como os da revista Microhobby.

Assim, colocaremos algumas regras, que entrarão em vigor a partir desta publicação. As colaborações recebidas anteriormente serão analisadas, uma a uma, e entraremos em contato com os autores oportunamente.

Assim, ficam estabelecidas as seguintes regras:

a) Os autores que enviarem colaborações para Microhobby aceitam estas regras em carta anexa ao artigo.

b) Uma vez recebida a colaboração, fica implícito que seu autor autoriza a publicação de seu artigo.

c) Os artigos passarão por uma triagem e os autores dos artigos aceitos receberão uma comunicação onde constará: valor da remuneração estabelecido pela redação, segundo seus critérios; um contrato de seção de direitos autorais que deverá ser assinado e devolvido à redação dentro de, no máximo, 10 dias. Qualquer discordância dos termos deverá ser comunicada neste período.

d) O pagamento será efetuado após a publicação do artigo, desde que o contrato tenha sido assinado. Caso o artigo já tiver sido publicado, o autor poderá assinar o contrato no momento do pagamento.

e) Os artigos não aceitos serão devolvidos aos autores (inclusive material anexo: fitas e listagens).

f) Os artigos e programas remunerados serão considerados propriedade da Micromega PMD Ltda., podendo esta fazer o uso que lhe convier (inclusive não publicá-lo).

g) As colaborações poderão ser:

1) Artigo sobre assuntos relativos à Informática de um modo geral;

2) Artigos sobre computadores de uma das seguintes linhas: TK 83/85, TK 2000, Apple, TRS-80 ou seus compatíveis.

3) Programas para estas mesmas linhas. Junto com a listagem do programa deverá vir um texto explicativo sobre o funcionamento do mesmo, utilização, detalhamento, ou o que mais o autor achar conveniente. O computador utilizado deverá estar claramente indicado.

h) Programas acompanhados por fitas serão mais facilmente analisados.

i) O texto dos artigos ou programas deverão vir datilografados ou, pelo menos, escritos com letra legível.

j) Os autores asseguram a originalidade do texto e dos programas. As fontes deverão ser citadas.

Estes são os regulamentos. Todavia, quando seu autor manifestar-se expressamente por carta, poderemos estudar alterações, desde que sejam convenientes para ambas as partes.

COME-COBRA

PROGRAMA

Gustavo Egídio de Almeida

Você é uma cobra que está perdida em um labirinto. Ao caminhar, você terá que comer pequenos alimentos depositados no solo (.) e, por trás de seu percurso, será deixado um grande rastro (:) (figura 1).

Porém, as coisas não são tão simples assim e você estará sendo incansavelmente perseguido por um imenso e aterrorizante monstro (§).

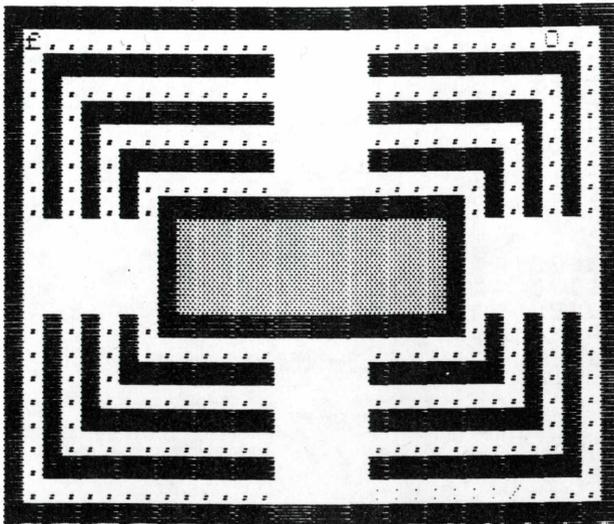
No interior do labirinto, será encontrado um alimento de maiores propriedades nutritivas (o) e, caso você o coma, ganhará mais pontos.

Os movimentos, tantos do monstro como os seus (da cobra), são feitos através de POKEs na tela, ao invés de usarem-se PRINTs, possibilitando um aumento na velocidade do jogo.

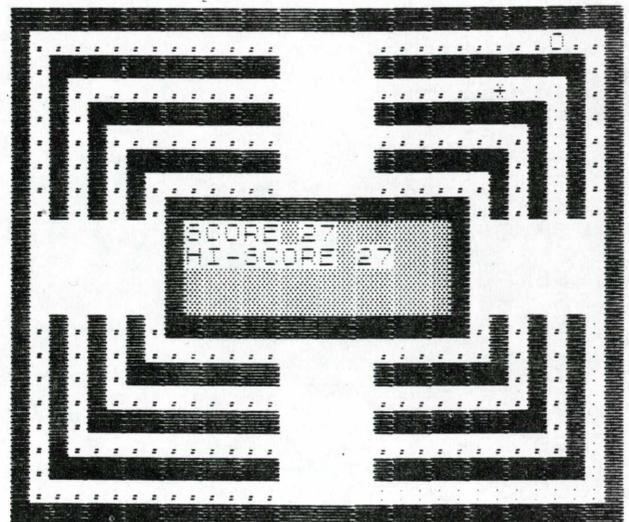
Para mover-se no interior do labirinto, use as teclas 5, 6, 7, 8 ou o joystick. Você deve estar pensando que a movimentação destes jogos é feita simplesmente pressionando qualquer uma destas teclas, no momento em que quiser, para mover o símbolo (/). Porém, engana-se. Você, ou seja, a cobra, só anda no sentido anti-horário e pode mudar de faixa apenas quando se encontra nas partes totalmente brancas do labirinto.

Com o monstro (§) já ocorre o inverso: ele percorre o sentido horário do labirinto e, a qualquer momento, poderá pegá-lo de frente. Caso você seja pego, ocorrerão piscadas na tela, implicando no final do jogo e surgirá seu placar e o recorde do jogo (figura 2).

Caso você consiga comer todos os pontos, uma nova etapa será iniciada, só que, desta vez, as coisas se inverterão, ou seja, você terá agora que comer (:) ao invés de (.). Esta é denominada a segunda fase do jogo.

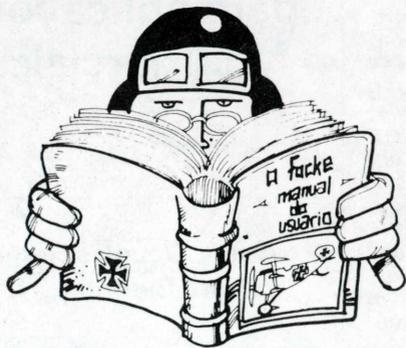
Na listagem, apresentada a seguir, os gráficos usados nas linhas de impressão devem ser rigorosamente os mesmos.

A linha 1 REM da listagem deve conter 20 caracteres quaisquer. Ao rodar o programa, esta linha se carregará automaticamente com os códigos decimais e o jogo se iniciará logo em seguida.

Boa sorte! . . .

Memória ocupada: 4380 bytes

```
   1 REM E£RND  F7 SAVE TAN LEN
 ?/ PAUSE
   2 GOSUB 1000
   3 SLOW
   4 LET H1=0
   5 LET S=0
   6 LET S1=0
   8 RAND
  10 PRINT "
  20 PRINT ". . . . . . . . . . . .
  25 LET O=0
  30 PRINT ".
  40 PRINT ". . . . . . . . . . .
  50 PRINT ". .
  60 PRINT ". . . . . . . . .
  70 PRINT ". . .
  80 PRINT ". . . . . . . .
  90 PRINT ". . . .
```

```
█████.█.█.█.█"
 100 FOR A=1 TO 4
 110 PRINT "█";TAB 8;"█▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒█";TAB 31;"█"
 120 NEXT A
 130 PRINT "█.█.█.█.███████████████████
█████.█.█.█.█"
 140 PRINT "█.█.█.█.........          .
.....█.█.█.█"
 150 PRINT "█.█.█.█████████████        █
███████.█.█.█"
 160 PRINT "█.█.█...........           .
.......█.█.█"
 170 PRINT "█.█.███████████████        █
█████████.█.█"
 180 PRINT "█.█............            .
.........█.█"
 190 PRINT "█.████████████████        █
███████████.█"
 200 PRINT "█..............            .
...........█"
 202 LET V=216
 205 LET G=14
 207 LET H=27
 209 LET V1=V
 210 PRINT "███████████████████████████████
█████████████"
 215 LET A1=PEEK 16396+256*PEEK
16397
 220 LET B1=0
 225 LET LA=1
 230 LET A=A1+678
 235 LET LB=1
 240 LET B=A1+299
 245 LET A2=0
 250 LET C=1
 252 LET O=A1+INT (RND*660)
 254 IF PEEK O<>27 AND PEEK O<>1
4 THEN GOTO 252
 255 IF PEEK O=H THEN LET V1=V1-
1
 256 POKE O,52
 257 LET V=V-1
 258 IF S>1 THEN RETURN
 260 LET D=-33
 270 IF A2=H THEN LET S=S+1
 280 IF PEEK (A+C)=128 THEN GOSU
B 400
 290 POKE A,G*((A2=H)+(A2=G))
 292 IF S=V1 THEN GOSUB 900
 295 IF INKEY$<>"" AND PEEK (A+C
)=0 THEN GOSUB 700
 300 LET A=A+C
 301 IF A=O THEN LET S1=S1+5
 302 IF A=O THEN GOSUB 252
 304 LET A2=PEEK A
 305 IF PEEK A=12 THEN GOTO 500
 310 POKE A,24
 320 IF PEEK (B+D)=128 THEN GOSU
B 450
 330 POKE B,B1
 335 IF B1=0 AND Q=0 AND LA<>LB
THEN GOSUB 800
 337 IF B1<>0 THEN LET Q=0
 340 LET B=B+D
 345 IF PEEK B=24 THEN GOTO 500
 350 LET B1=PEEK B
 360 POKE B,12
 370 GOTO 270
 400 LET X=0
 402 IF C=1 THEN LET X=-33
 405 IF X=-33 THEN GOTO 435
 410 IF C=-33 THEN LET X=-1
 415 IF X=-1 THEN GOTO 435
 420 IF C=-1 THEN LET X=33
 425 IF X=33 THEN GOTO 435
 430 IF C=33 THEN LET X=1
 435 LET C=X
 440 RETURN
 450 LET Y=0
 452 IF D=-33 THEN LET Y=1
 455 IF Y=-33 THEN GOTO 485
 460 IF D=1 THEN LET Y=33
 465 IF Y=33 THEN GOTO 485
 470 IF D=33 THEN LET Y=-1
 475 IF Y=-1 THEN GOTO 485
 480 IF D=-1 THEN LET Y=-33
 485 LET D=Y
 490 RETURN
 500 POKE A,23
 510 FOR M=1 TO 26
 520 RAND USR 16514
 530 NEXT M
 585 LET S=S+S1
 590 SLOW
 600 PRINT AT 9,9;"SCORE▒";S
 605 IF H1<S THEN LET H1=S
 610 PRINT TAB 9;"HI-SCORE▒";H1
 620 PAUSE 35000
 630 CLS
 640 GOTO 5
 700 LET A3=A
 705 LET A$=INKEY$
 710 LET A=A+(((INKEY$="8")-(INK
EY$="5"))*(ABS C=33)+((INKEY$="6
")-(INKEY$="7"))*33*(ABS C=1))*2
 720 IF A>A1+726 OR A<A1 OR PEEK
 A<>0 THEN LET A=A3
 730 IF A=A3 THEN RETURN
 740 LET L5=LA+(C=-1)*(A$="6")+(
C=1)*(A$="7")+(C=-33)*(A$="5")+(
C=33)*(A$="8")
 750 IF L5=LA THEN LET L5=LA-1
 755 LET LA=L5
 760 RETURN
 800 LET Q=1
 810 LET D1=D
 820 GOSUB 450
 830 LET D2=D
 840 LET D=D1
 850 LET W=LA-LB
 860 IF W>1 THEN LET W=1
 870 IF W<-1 THEN LET W=-1
 875 LET LB=LB+W
 880 LET B=B+W*D2*2
 890 RETURN
 900 LET S1=S1+S
 910 LET S=0
 920 LET G=H
 930 IF H<>G THEN GOTO 950
 940 LET H=14
 950 LET V1=V
 960 RETURN
1000 LET M$="042 012 064 006 023
 043 035 126 254 118 032 003 016
 246 201 198 128 119 024 242"
1010 FAST
1020 FOR M=16514 TO 16533
1030 POKE M,VAL M$( TO 3)
1040 LET M$=M$(5 TO )
1050 NEXT M
1060 RETURN
1100 SAVE "COME/COBR█"
1105 SLOW
1110 PRINT TAB 10;"COME-COBRA"
1120 PRINT AT 8,0;"VOCE E█ A COB
RA(/) E DEVE COMER O MAIOR NUMER
O POSSIVEL DE:"
1130 PRINT "(.)>FASE-█"
1140 PRINT "(:)>FASE-£"
1150 PRINT "CASO COMA (O) GANHAR
A MAIS PTOS"
1160 PRINT "O MONSTRO E█ REPRESE
NTADO POR(£)"
1170 PRINT AT 21,0;"TECLE 0 PARA
 COMECAR"
1180 IF INKEY$<>"0" THEN GOTO 11
80
1190 CLS
1200 RUN
```

LIVROS

Tecnologia e Soberania Nacional

Ana Lúcia de Alcântara

Autor: Waldimir Pirró Longo

Editora Nobel e PROMOCET — Promoção de Ciência e Tecnologia

Através da iniciativa das duas instituições surgiu este livro que, segundo a editora, pretende situar a tecnologia como fator estratégico de bens e serviços e como ordenadora da distribuição da riqueza e do trabalho no mundo atual.

A PROMOCET, uma instituição voltada à promoção do desenvolvimento científico e tecnológico, e a Editora Nobel publicam o texto de Pirró com o intuito de fornecer ao público "um entendimento claro sobre as questões centrais que envolvem a definição de um modelo econômico-tecnológico para o Brasil".

O livro

"Tecnologia é o conjunto organizado de todos os conhecimentos empregados na produção e comercialização de bens e serviços e o seu domínio é o que permite a elaboração das instruções necessárias para a produção dos mesmos". A partir deste pressuposto, o autor desenvolve todo o texto do livro, mostrando desde os conceitos básicos sobre ciência e tecnologia até os parâmetros que envolvem o poder nacional e a soberania de um país.

O autor afirma: as tecnologias são interdependentes, existindo as tecnologias correlatas (central e paralela). Este fato, segundo Longo, traz conseqüências sérias a serem consideradas; uma delas é a incapacidade de um país — cujo setor produtivo é fortemente desnacionalizado e no qual as empresas nacionais são dependentes de tecnologias importadas — levar avante uma política de independência do exterior em algum setor estratégico isolado. Como exemplo, o autor cita a eletrônica: "ao se permitir a total desnacionalização da indústria de componentes, torna-se extremamente difícil atingir razoável independência em algum produto que dependa da eletrônica".

Assim, Waldimir Pirró Longo desenvolve sua argumentação. Em características da tecnologia e sua comercialização, ele ressalta a importância da tecnologia explícita — que se encontra acumulada em pessoas, sob a forma de conhecimentos intelectuais e habilidades manuais — e afirma que a transferência de tecnologia geralmente está ligada a ela (uma vez que, em geral, é formalizada através de contratos): ilustrando ainda que realmente ocorre uma venda, na qual, quase sempre, o vendedor esconde os conhecimentos e vende as instruções.

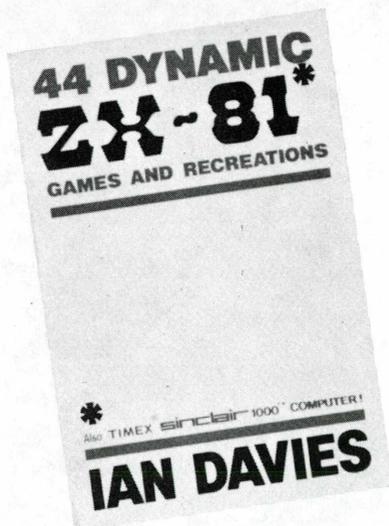
Sobre este tema (transferência de tecnologia), Longo dedica um capítulo especial mostrando que "a verdadeira transferência só ocorre quando o receptor absorve o conjunto de conhecimentos que lhe permite inovar". E complementa: "a transferência não se solidifica se o comprador não dominar os conhecimentos envolvidos a ponto de ficar em condições de criar nova tecnologia".

"Multinacionais e tecnologia, investimentos em P&D, tecnologia na conjuntura estratégica mundial" são capítulos que se preocupam em demonstrar a importância do domínio do conhecimento tecnológico e as conseqüências para os setores social e econômico. Num dos capítulos, Pirró Longo ilustra: "a formação de uma sociedade científica e tecnologicamente capaz começa no ensino de 1º e 2º graus, nos quais o aprendizado das ciências deve ser conduzido com competência e seriedade" e conclui afirmando que o raciocínio independente e a criatividade devem ser exercitados e perseguidos como metas prioritárias na formação dos alunos".

O autor soube finalizar o livro de forma bastante expressiva dividindo dois ítens importantíssimos da discussão sobre tecnologia, poder e independência econômica. Na primeira etapa, vislumbrou-se os efeitos da dependência tecnológica nos diversos componentes do Poder nacional — nos campos psicossocial, econômico, militar e político — e finalizando, o autor apresenta aspectos que, para ele, são importantes para a elaboração de um plano governamental no desenvolvimento da ciência e da tecnologia:

— "Para que um plano de desenvolvimento de C&T tenha conseqüências, é preciso que ele seja precedido de um planejamento explícito para o setor produtivo, em que fique claro qual é o espaço reservado para a criatividade local, onde é para valer o esforço da inteligência nacional".

44 Dynamic ZX-81 Games and Recreations

Autor: Ian Davies
Editora Prentice-Hall

Como o próprio título afirma, o livro é constituído de 44 programas-jogos em BASIC para micros ZX-81 e também os compatíveis com os TKs.

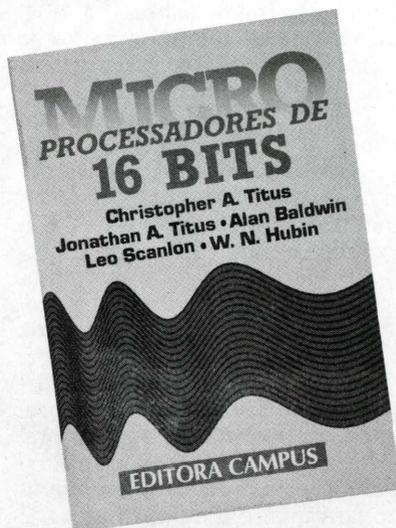
Editado em inglês (não existe tradução em português no Brasil), "44 Dynamic ZX-81" destina-se, como o próprio autor afirma no prefácio, àquelas pessoas que compraram um micro pessoal com o intuito de ter às mãos um videogame; para professores que adquiriram o micro para suas aplicações educacionais (para isto o livro traz programas como "quadratic equations, elementary arithmetic") e outros usuários.

Cada programa é apresentado em seis seções ou subdivisões: o que o programa faz e como usá-lo; modificações que podem ser feitas; explicações sobre técnicas especiais utilizadas no programa; listagem e descrição das variáveis e das sub-rotinas.

O livro, no final, traz dois apêndices que falam sobre os termos utilizados e um capítulo dedicado à "programação eficiente" (efficient programming).

"44 Dynamic ZX-81" pode ser encontrado na Litec — Livraria Técnica em São Paulo, através do telefone: (021) 220.8983/221.1921 ou na representação da Prentice-Hall no Brasil, através do telefone (021) 285.3275. A.L.A.

Microprocessadores de 16 bits

Diversos autores
Editora Campos

Comparar alguns processadores de 16 bits (8086 da Intel, o Z8001/Z-Zilog, o 9900 da Texas, o LSI-11/DEC, o 6800/Motorola e o 16000 da National) entre si, através de benckmarks simples, feitos por autores diferentes. Estes são os principais objetivos do livro.

O básico sobre microprocessadores foi deixado de lado sendo focalizado apenas os principais termos técnicos utilizados, de maneira abrangente (organização interna do microprocessador, conjunto de instruções, sistemas de desenvolvimento, etc.), já que os autores levaram em cosideração que os leitores do livro já possuem o conhecimento necessário do vocabulário utilizado na área (EPROM, RAM, ROM, interrupção, registro e afins).

O livro é dividido em sete capítulos e apêndices que trazem os benckmarks de classificação pelo método da Bolha, de pesquisa de strings e tabelas de consultas.

As características, histórico, software, benckmarks, hardware do chip dos vários microprocessadores são abordados em capítulos separados, através de gráficos, tabelas e ilustrações. **A.L.A.**

BASIC para aplicações comerciais

Autor: Douglas Hergert
Editora Campos

Este livro destina-se àqueles que querem iniciar-se na programação BASIC de microcomputadores seguindo uma orientação específica. O objetivo do autor é treinar profissionais da área comercial a "escreverem, lerem e depurarem programas em BASIC" para suas aplicações.

Dividido em sete capítulos e dois apêndices, "BASIC para aplicações comerciais" fala sobre diversas características do BASIC, enfocando programas que executam determinadas tarefas comerciais, ao mesmo tempo em que traz no final de cada capítulo, exercícios que orientam o leitor sobre os tópicos abordados.

Um detalhe que deve ser destacado é a preocupação em comparar as características do BASIC a outras linguagens ligadas ao comércio como FORTRAN, COBOL e PASCAL.

O livro esclarece as vantagens específicas oferecidas para cada uma destas linguagens e nos apêndices o autor apresenta programas em COBOL (faturas), em Pascal (um programa que emite relatórios mensais de vendas), em FORTRAN (uma versão do programa de depreciação apresentado em um dos capítulos), etc. **A.L.A.**

Chegou a mais alta patente em videogame.

Onyx Junior. Uma verdadeira batalha de emoções.

Prepare-se. Dentro de sua própria casa, você vai perseguir e abater mísseis e tanques de guerra. Seus jatos escaparão por milímetros dos potentes canhões inimigos, planetas explodirão em chamas. Você vai conhecer, nas cores mais dramáticas, na maior nitidez, as sensações de uma verdadeira batalha.

Entregue-se. Você simplesmente não vai resistir a fantástica aventura que é ter um Onyx Junior.

O mais moderno, o mais completo videogame que você já viu (o único neste sistema que tem pause),com mais de 300 jogos diferentes: o Onyx Junior usa todos os cartuchos da Linha Atari®.

Apresente-se. A mais alta patente em videogame espera você para um encontro inesquecível.

MICRODIGITAL

CERTIFICADO ESPECIAL DE RESERVA
VÁLIDO ATÉ 30.04.85

AUTORIZO PELO PRESENTE MINHA ☐ ASSINATUHA INICIAL ☐ RENOVAÇÃO ☐ ALTERAÇÃO DE ENDEREÇO DA REVISTA MICROHOBBY (12 EDIÇÕES).

ANEXO REMETO ☐ VALE POSTAL Nº ______________ ☐ CHEQUE Nº ______________ BCO. ______________

NOMINAL À MICROMEGA P.M.D. LTDA. NO VALOR DE CR$ ______________________

NOME ______________________________________

ENDEREÇO ______________________________________

CIDADE ______________ BAIRRO ______________ CEP ______________

FONE ______________ ESTADO ______________

PROMOÇÃO SENSACIONAL
VOCÊ ECONOMIZA
Cr$ 18.000/21.000

PREÇOS: ASSINATURA: Cr$ 30.000
RENOVAÇÃO: Cr$ 27.000

Em caso de renovação de assinatura colar neste campo a etiqueta de endereçamento atual.

Micromega P.M.D. Ltda. • Av. Angélica, 2318 • 14 And. • São Paulo • Cep 01296
• Caixa Postal 54096 • Fone: 826-5001

MICROHOBBY

cada vez melhor!

A MICROHOBBY é uma revista altamente didática, destinada a programadores de vários níveis, do principiante ao hobbista mais ousado, que se aventure a programar em linguagem de máquina.
Receba em sua casa a revista que contém inúmeros programas, informações, dicas e tudo o que você precisa saber sobre microcomputadores e programação.
MICROHOBBY a revista que põe você em dia com a informática!

promoção especial para novos assinantes e renovações

Fazendo sua assinatura agora você além de manter o preço inalterado durante 12 edições, paga apenas 10 edições (Cr$ 30.000); e ainda ganha uma fita de brinde no valor de Cr$ 12.000 com 2 jogos. E um grande desconto, você ganha a preço de hoje Cr$ 18.000.
Renovando sua assinatura além das vantagens já citadas você ainda terá um desconto de Cr$ 3.000, ou seja sua renovação sairá por apenas Cr$ 27.000, você ganha Cr$ 21.000.
Assinando ou renovando agora você só tem a ganhar.

Assinando a revista MICROHOBBY, você recebe inteiramente grátis uma fita cassete contendo 2 jogos.
1 - O pouso do Barão Vermelho e Pac-hobby (TK83/85)
2 - Calendário perpétuo e Tatuzão (TK2000 e APPLE)

Qual a marca do seu micro? ______________________

MICROHOBBY

PEDIDO DE LIVROS E NÚMEROS ATRASADOS DA MICROHOBBY

(Preencher os ítens no verso e o cupom abaixo)

Sim, desejo receber os ítens assinalados no verso

NOME

ENDEREÇO

CEP

CIDADE

EST.

TOTAL DO PEDIDO: Cr$ ____________________

Enviar cheque nominal cruzado ou vale postal
à MICROMEGA P.M.D. Ltda. — Cx. Postal 54096
CEP 01296 — São Paulo — SP — Fone: 826-5001

Cheque nº ____________________

Bco. ____________________ ☐ Vale postal

Estou ciente de que aguardarei em média 30 dias para o recebimento dos produtos

DATA ______ / ______ / ______

____

assinatura